Research and Application of Target Tracking Algorithm in Ship Safety

Bailin Lin\textsuperscript{a} and Zheng’an Xiao\textsuperscript{b}

School of Physics and Mechanical&Electrical Engineering, Hubei University of Education, Wuhan 430205, China

\textsuperscript{a}home12345@163.com; \textsuperscript{b}285106243@qq.com

Abstract. Target tracking is often used in tasks such as video surveillance. Its purpose is to obtain the target of interest in the sequence image accurately, robustly and in real time, and to establish the connection between the moving objects in each image. In recent years, with the continuous development of the global shipping economy, ships have gradually developed into large-scale and high-speed, and how to ensure the safety of ships sailing in these sports has become an important issue. Based on the principle of particle filter and mean shift algorithm, this paper takes the moving ship as the research object and uses the polynomial fitting method to analyse the performance of the two algorithms. A large number of experimental results show that the mean shift algorithm has higher accuracy, and the particle filter method has better real-time performance. And as the number of particles increases, the accuracy of particle filtering will gradually increase.

1. Introduction

\begin{figure}[h]
  \centering
  \includegraphics[width=0.5\textwidth]{figure1.png}
  \caption{Block Diagram of Ship Automatic Tracking System.}
  \label{fig:1}
\end{figure}

The purpose of video target tracking [1] is to analyse the video sequence captured by the sensor and correlate the same moving target in different frames of the image sequence to obtain the complete motion track of each moving target. The essence of target tracking is to analyse the video sequence captured by the image sensor and calculate the position, size and movement speed of the target in each frame of the image. As a main branch of the field of computer vision, the research of video target tracking method is increasingly applied to security, intelligent video surveillance, intelligent traffic
management, military robot vision and other fields. This article takes sports ship target tracking as an example to carry out relevant algorithms and experimental research. The system that is formed is shown in Figure 1. The shipping monitoring system collects images or video and transmits it to image processing equipment and then detects the target according to ship template matching and Track your goals.

In this paper, the sports ship is taken as the tracking target. Each frame of image in the video is taken as the object of processing and the image is enhanced. Then, the target ship is tracked by Kalman and particle filter respectively.

2. Algorithm principle analysis

2.1. Mean Shift algorithm

Mean Shift [2] is a typical algorithm for the offset vector. In the target tracking, the probability density function is used to iteratively search for the target colour feature to track the target. Now commonly referred to as Mean Shift algorithm belongs to the search and match category in the tracking. According to the maximum similarity of the tracking template, the mean value of the current point and the mean value of the shifted point are automatically found in the neighbouring area. This is a relatively simple density estimation model. Constant search until the calculated offset point is the last pixel to scan the entire image [3]. The specific process of the algorithm is as follows.

1) The target model of the initial frame

In the target search window space, the interval is divided into K equal intervals, and the central pixel coordinate of the search window in the initial frame is set as the coordinate of the i-th pixel in the image space area. In addition, \( k(\|x\|^2) \) is used to represent the kernel function selected during processing and the window width is \( h \). Then the probability of the u-th eigenvalue can be expressed by formula (1).

\[
\hat{q} = C \sum_{i=1}^{n} k\left(\frac{X_i - x}{h}\right) \phi[b(x - u)]
\]

(1)

In (1), the membership of the color corresponding to the characteristic value u in the coordinates can be judged by b. In the color space, C is the constant coefficient to make the probability normalized.

\[
\sum_{i=1}^{n} \hat{q} = 1
\]

(2)

2) Current frame model

According to the establishment of the initial frame model, the probability of the u-th feature value of the search window in the current frame can also be expressed as:

\[
\hat{p}(y) = C_h \sum_{i=1}^{n} k\left(\frac{Y_i - y}{h}\right) \phi[b(x - u)]
\]

(3)

In equation (3), the central pixel coordinates corresponding to the current frame search window in the initial frame model correspond to C in equation (1).

3) Similarity function

The similarity between the target model and the current frame model in the initial frame can be described by a similarity function, which is defined as (4):

\[
\rho(\hat{p}(y), \hat{q}) = \frac{1}{2} \sum_{u=1}^{m} \sqrt{\hat{p}(y)\hat{q}\hat{u}} + \frac{C_h}{2} \sum_{i=1}^{n} w_i k\left(\frac{Y_i - y}{h}\right)^2
\]

(4)
Where  \( w_i = \sum_{i=1}^{m} \frac{\hat{\varphi}_i}{\hat{\varphi}_i - \hat{\varphi}_i(y_0)} \delta[p(x_i - u)] \).

In the feature interval, the desired Mean Shift vector can be obtained when the similarity function  \( \rho(y) \) is at the maximum value [4]:

\[
m_{h,G}(y) = y_1 - y_0 = \left[ \frac{\sum_{i=1}^{\infty} x_i w_i g \left( \frac{\|\hat{y} - x_i\|^2}{h} \right)}{\sum_{i=1}^{\infty} w_i g \left( \frac{\|\hat{y}_0 - x_i\|^2}{h} \right)} \right] - y_0
\]

(5)

2.2. Particle Filter Fundamentals

The basic idea of particle filtering is to firstly generate a set of random samples in the state space based on the empirical conditional distribution of the system state vectors. These samples are called particles. Then the weights and positions of the particles are continuously adjusted according to the measurements, and the particles are adjusted. The information corrects the initial distribution of experience conditions. The essence of this approach is to approximate the associated probability distributions using discrete random measures consisting of particles and their weights and recursively update the discrete random densities according to the algorithm. When the sample size is large, this Monte Carlo description approximates the true nonlinear stochastic system of the state variable. The precision can be approximated to the optimal estimate. It is a very effective nonlinear filtering technique [4].

Particle filtering, also known as sequential Monte Carlo method (SMC), refers to finding a series of random samples that can approximately express the probability density function  \( p(x_k | z_k) \) in the state space and replacing the integral operation with an arithmetic mean value to obtain the state minimum variance of the process of estimation. These random samples are called "particles".

Combining the SIS with the particle resampling method, a complete particle filter algorithm can be obtained. Specific steps are as follows:

1) Initialization
Based on the distribution of, random sample in the vicinity of  \( \) to obtain a series of particles.

2) SIS
Usually the importance distribution  \( q(x_k | x_{0:k-1}, z_{1:k}) \) can be simplified to  \( p(x_k | x_{k-1}) \) and N particles  \( x_k^i \propto p(x_k | x_{k-1}) \) can be sampled.

3) Calculate weights
Calculate the weight of each particle  \( w_k^i \) and normalize it  \( w_k^i = w_k^i / \sum_{i=1}^{N} w_k^i \).

4) Posterior probability estimation
Output a set of weighted particles  \( \{x_k^i, w_k^i \}_{i=1,2,...,N} \). According to the weighted average or maximum posterior probability of the particle, the posterior probability estimate of the current time  \( k \) can be obtained.

5) Particle resampling
According to the particle weights, the sample set is re-sampled. The large-weighted particles are divided into multiple particles while the small-weighted particles are deleted and the new N particles are obtained.

6) State transfer
When  \( k+1 \) arrives, record the observations and repeat from 2 to 6.
3. Sports ship tracking experiment

3.1. Mean Shift tracking algorithm

The traditional Mean Shift algorithm was used for target tracking in the experiment. The first step is the initialization of the target tracking, which can be used to obtain the circumscribed rectangle of the initial target that needs to be tracked. However, the main research focus of this paper is not on the target detection module but to track the algorithm, so the manual selection method by the mouse is selected. Then the histogram distribution of the search window weighted by the kernel function is calculated and with the same method the histogram distribution of the corresponding window of the Nth frame is calculated. Based on the principle of maximum similarity between the two target template distributions, the search window is moved in the direction in which the density increases the most, and the true position of the target is obtained. The tracking steps are as follows.

1) Calculate the probability density \( q_u \) of the target template, the target estimated position \( y_0 \) and the nuclear window width \( h \).

2) Using the target position \( y_0 \) of the initial frame to calculate the candidate target template \( \{p_u(y_0)\}_{u=1...m} \).

3) Calculate the weight value of each point in the current window.

4) Calculate the new position of the target:

\[
\begin{align*}
\sum_{i=1}^{m} x_i w_i g \left( \frac{y_i - y_0}{h} \right) \\
\sum_{i=1}^{m} w_i g \left( \frac{x_i - y_0}{h} \right)
\end{align*}
\]

(6)

The Mean Shift algorithm continuously iterates and finally finds the optimal position of the target in the image sequence.

In the experiment, the tracking window is first determined and the tracking target model of the initial frame is set. Select the target to be tracked in the first frame, and then frame a rectangle containing the ship’s target, that is, the ship feature search window, also called the tracked target area. Cut the area and the result is shown in Figure 2. Observe the record tracking results after the end of the trace, as shown in Figure 3 to 6.

It can be seen from the experimental results that the two ships are staggered at 50 frames. The tracking frame is only slightly offset at a video around the 100th frame in the middle, but then the tracking frame is restored again until the final 273th frame. Observing the tracking trajectory basically conforms to the motion situation, but the upper and lower hopping occurs. On the one hand, there is tracking error, but the main reason is that the camera does not use a tripod during video capture and the image is shaken, so it is in an acceptable error range.

![Figure 2. Image Selection.](image-url)
3.2. Ship tracking based on particle filter

Particles [5] used in target tracking usually refer to target status information, including target position, size, color, and motion information [6]. The specific implementation steps are as follows.

1) Initialization

The initial position of the target $X_0$ is selected and $N$ particles are randomly distributed near $X_0$.

2) Get particle weight

Assuming the system is in a state such as $X_1, X_2, ..., X_n$, where $X_i, i = 1, 2, ..., n$, which we call a particle, represents a particle-centric image block. For each particle, the position, color, edge, and other information together form the current state of the moment. $Z$ is the observed value of the system at this time, and $p(z_k|X_k)$ is obtained based on historical information. To estimate the exact position of the target $X$, it is necessary to operate on the particles. Each particle $X_i$ corresponds to a rectangular frame, and the color in the frame is counted to obtain a 16-dimensional color histogram vector $H_i$. Each color histogram has 16 bins containing 16 grayscale values (0~255/16). After the statistics, the $H_i$ is normalized so that the sum of the squares of each bin is 1, which is a 16-dimensional unit vector.

The weight $W_i$ of each $H_i$ can be understood as the inner product of the color histograms $H$ and $H_i$ of the target template, that is, the quantities of each bin are multiplied and added. This represents the similarity between the image region represented by the current particle and the target template. The higher the weight, the more similar the particle is to the target. The weights $W_1, W_2, ..., W_n$ of $X_1, X_2, ..., X_n$ are calculated from $W_i = \exp(10^* <H, H_i>)$, where $<H, H_i>$ represents the inner product operation.

3) Predict the current position

From the weighted average of the obtained particles $X_1, X_2, ..., X_n$ and the corresponding weight
\( W_1, W_2, \ldots, W_n \), the system state can be estimated: 
\[
X = X_1 * W_1 + X_2 * W_2 + \ldots + X_n * W_n ,
\]
where \( X_i \) represents the location information of the center of the area. The resulting \( X \) is approximately equal to the posterior probability estimate and can be used to predict the current position of the target.

4) Particle resampling

As the number of video dumps increases, the weight will be more and more concentrated on a small number of particles, and most other particles have very small weights. Before entering the next iteration, the particle weights are redistributed to make them all the same. Therefore, the particles need to be re-sampled, i.e., \( N \) particles \( X'_1, X'_2, \ldots, X'_n \) are regenerated. The regenerated particles all come from the original particle set \( X_1, X_2, \ldots, X_n \), in which particles with a large weight generate more new particles, and particles with smaller weights have fewer new particles.

5) State transfer

At the next moment, the particles are updated from \( X'_i \) to \( X''_i \). Since the probability that the system state is transferred transition from \( X'_i \) to \( X''_i \) according to the state matrix is \( p(x_k | x_{k-1}) \), the probability of changing from \( X'_i \) to \( X''_i \) is equal to \( p(x_k | x_{k-1}) \).

6) Repeat 2 to 5

In the experiment, the number of particles was selected as 200. The particle filter algorithm of this paper also adopts the method of manually selecting the tracking target. First display the initial frame and use the marquee to select a rectangle containing the ship's target, which is the ship feature search window. The feature search box is cut out as far as possible from the edge of the ship and cut out with the right mouse button. The cut is performed using the [temp, rect]=imcrop(I) format, and the image matrix of the ship model is stored in the temp variable. The histogram is then calculated for this variable. The initial position coordinates of the ship are stored in the variable rect and for calculating the ship center position coordinates.

![Figure 7. Frame 20.](image1)

![Figure 8. Frame 53.](image2)

![Figure 9. Frame 70.](image3)

![Figure 10. Frame 80.](image4)

After the end of the tracking, the tracking results are observed. The results are shown in Figure 7 to Figure 10.
4. Conclusion
In this paper, the tracking accuracy and real-time performance of the two algorithms are compared. It is found that the accuracy of the particle filter tracking algorithm is lower than that of the mean shift algorithm, but the real-time performance is better. Then, the influence of particle number on particle filter is studied. It is found that as the number of particles increases, the accuracy of tracking increases, but the running time becomes longer and the real-time performance deteriorates.

Acknowledgments
This work was supported by Hubei Natural Science Foundation 2016CFB681.

References
[1] Bourque A E, Bedwani S, Carrier J F, et al. Particle Filter–Based Target Tracking Algorithm for Magnetic Resonance–Guided Respiratory Compensation: Robustness and Accuracy Assessment[J]. International Journal of Radiation Oncology* Biology* Physics, 2018, 100(2): 325-334.
[2] Comaniciu D, Meer P. Mean shift: A robust approach toward feature space analysis[J]. IEEE Transactions on pattern analysis and machine intelligence, 2002, 24(5): 603-619.
[3] Cheng Y. Mean shift, mode seeking, and clustering[J]. IEEE transactions on pattern analysis and machine intelligence, 1995, 17(8): 790-799.
[4] Collins R T. Mean-shift blob tracking through scale space[C]. Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on. IEEE, 2003, 2: II-234.
[5] Hans R. Künsch. Particle filters[J]. Statistics, 2013
[6] Pitt M K, Shephard N. Filtering via simulation: Auxiliary particle filters[J]. Journal of the American statistical association, 1999, 94(446): 590-599.