Research on Video Image Tracking Method Based on Kernel Density Estimation

Ya Zhang[1, a, *1], YuanyuanXu [1], Yan Zhang [1]

[1 *]School of Electronic Information and Artificial Intelligence,YiBin Vocational And Technical College, China

ABSTRACT

Starting from the kernel density estimation method, this paper uses the similarity of color distribution density function to track video image target under the framework of KPF filter. By simplifying the system model and mean shift algorithm, fewer particles are used to improve the tracking accuracy and reduce the calculation cost. The validity of the method is verified by experiments.

Keywords: Kernel function; Density function estimation; Particle filter; Target tracking.

1. INTRODUCTION

Target tracking has always been an active research area in the field of computer vision[1]. It has many potential applications such as: robot control, human-machine interface, video communications, or surveillance[2]. Traditional particle filters sample observation data, and the resulting samples form particle swarms to approximate the posterior probability density of the target state to determine the location of the object. Therefore, a large number of particles is usually required to correctly estimate the posterior probability density [3,4].

Chang et al. Put up a nuclear -based particle filtering method[5]. This method belongs to the non -parameter method. Use the kernel function to estimate the density (KDE), and then estimate the post-test density gradient to move the sampl2e particles along the gradient direction to the local maximum value of the estimated density. Through such a mean flat transfer algorithm, the particles are allocated to the position closer to the maximum likelihood probability[6]. With a small amount of particles, the target position can be covered, the amount of calculation is reduced, and the estimation accuracy is improved[7]. However, the latest observations in this method are only used for weighted steps, not sampling steps. If the initial particles fail to cover the target position well, then the average value of the particles can not be able to cover the target area well, which will also cause the number of MS iterations to increase greatly.

KPF (Kalman particle filter) is a particle filter that includes the principle of Kalman filtering[8,9]. Since the state is updated with the latest observations during the filtering step of the Kalman filter, the above-mentioned problems of conventional particle filtering are avoided[10,11]. This paper combines the KPF and KDE methods, uses the KDE method to estimate the color density distribution of the target, and uses the estimated target color posterior density as the importance distribution of KPF to achieve tracking of video targets. In density estimation, the mean shift algorithm is used to improve tracking accuracy.

2. COLOR LIKELIHOOD FUNCTION ESTIMATION

2.1 Color likelihood model

Color features are convenient for describing targets with changing shapes. The description of the target is relatively stable when the target rotates, changes in size, and is partially blocked. The color distribution of the target area is represented by a discretized color histogram. The levels of the three color channels of R, G, and B in the histogram bin (bin) are all 16 levels respectively, recorded as:$m = 16 \times 16 \times 16$

[1a, *] zhangya909@gmail.com

The International Conference Optoelectronic Information and Optical Engineering (OIOE2024), edited by Yang Yue, Lu Leng, Proc. of SPIE Vol. 13513, 1351315 · © 2025 SPIE · 0277-786X · © The Authors.
Published under a Creative Commons Attribution CC-BY 3.0 License · doi: 10.1117/12.3045450

The candidate target area is a rectangle, the coordinates of the pixel points in the area are $\{px_i\}_{i=1....n_{pix}}$,and assuming the centroid position is $Y = [x, y]^T$, then the color likelihood distribution in the candidate target area is:

$$\hat{p}_j(X) = \frac{\sum_{i=1}^{n_{pix}} k(\left\|\frac{Y-px_i}{h}\right\|^2)\delta[b(px_i)-j]}{\sum_{i=1}^{n_{pix}} k(\left\|\frac{Y-px_i}{h}\right\|^2)} \text{ , j=1....m} \qquad (1)$$

Among them: $k(\cdot)$is the kernel function, and the Epanechnikov kernel is selected here; h is the window width, and its value is set to 0.315; $\delta(\cdot)$is the Dirac function.

In the first frame of the video, the target template can be determined manually or morphologically, and the reference color likelihood distribution $\hat{q} = \{q_j\}_{j=1....m}$ of the template can also be calculated by equation (1).

The Bhattacharryya coefficients of the two color likelihood distributions are:

$$\rho(\hat{p}(X), \hat{q}) = \sum_{j=1}^{m} \sqrt{\hat{p}_j(X)\hat{q}_j} \qquad (2)$$

The similarity of color distribution is defined as:

$$D = \sqrt{1 - \rho(\hat{p}(X), \hat{q})} \qquad (3)$$

The color likelihood function is:

$$p(Z^c|X) = \frac{1}{\sqrt{2\pi}\sigma_c} exp\,(-\frac{D^2}{2\sigma_c^2}) \qquad (4)$$

Where: σ_c is the variance of the color Gaussian distribution.

2.2 Mean shift

According to the formula (1), the color likelihood distribution in the candidate area is obtained, whereby the sample point can be moved along the gradient direction of the distribution to the local maximum value of the color likelihood distribution [5,12]. Assume that the position of the center of mass of the predicted target is Y_0 and the pixel coordinates of the candidate target area are $\{px_i\}_{i=1....n_{pix}}$, then the steps of the mean shift algorithm are:

Search along the gradient direction of the color likelihood distribution:

$$Y_1 = \frac{\sum_{i=1}^{n_{pix}} k'(\left\|\frac{px_i-Y_1}{h}\right\|^2)\cdot px_i\cdot w(px_i)}{\sum_{i=1}^{n_{pix}} k'(\left\|\frac{px_i-Y_1}{h}\right\|^2)\cdot w(px_i)}. \qquad (5)$$

$w(px_i)$ is the weight:

$$w(px_i) = \sum_{j=1}^{m} \sqrt{\frac{q_j}{p_j(Y_0)}}\delta[b(px_i) - j]. \qquad (6)$$

If $\|Y_1 - Y_0\| < \varepsilon$, then end the loop; otherwise $Y_0 = Y_1$, jump to step 1.

3. KPF TARGET TRACKING ALGORITHM BASED ON KERNEL DENSITY ESTIMATION

KPF is a particle filter. The probability distribution of the state of the tracking object is approximated by a group of particles $u_k^{(i)} = \{x_k^{(i)},\ w_k^{(i)}\}_{i=1....N}$, where $x_k^{(i)}$ represents the sampling of the state at k time and $w_k^{(i)}$ is its corresponding weight. In this article, state is considered as the location of an object. Particles propagate according to the target motion mode[13,14]. At time k, after obtaining the observation value z_k, the probability density of the tracked target position is expressed as the posterior density $p(x_k|z_k)$. This density function is difficult to express analytically in practice, and the kernel density function represented by equation (1) is often used for estimation [15].

3.1 Particle morphology in KPF

KPF follows the Kalman filter architecture, and its particle shape is divided into two stages: prediction and update, which are recorded as:

Prediction status: $\hat{X}_k = E[X_k|Z_k,\ldots,Z_1]$, corresponding variance: $\hat{P}_k = [(X_k-\hat{X}_k)(X_k-\hat{X}_k)^T]$

Update status: $\tilde{X}_k = E[X_k|Z_{k-1},\ldots,Z_1]$, corresponding variance: $\tilde{P}_k = [(X_k-\tilde{X}_k)(X_k-\tilde{X}_k)^T]$

Therefore, each particle in KPF actually consists of three parts, namely $u_k^{(i)} = \{X_k^{(i)}, P_k^{(i)}, w_k^{(i)}\}_{i=1\ldots N}$. The filtered output state is:

$$E[X_k] = \sum_{i=1}^{N} w_k^{(i)} \tilde{X}_k^{(i)} \qquad (7)$$

3.2 Target motion model

At time k, the observation vector of the target motion is $Z_k = [x_k, y_k]^T$, x_k and y_k are the x-coordinate and y-coordinate of the target in the image respectively. When the target makes random movements, the random walk motion model is used to describe its motion state:

$$X_k = X_{k-1} + \omega_k$$

$$Z_k = X_{k-1} + \upsilon_k \qquad (8)$$

Among them:$X_k = [x_k, y_k]^T$,ω_k and υ_k are process noise and observation noise respectively, which are independent of each other and obey Gaussian distribution. The corresponding variances are: $Q = \sigma_w^2 I$ and $R = \sigma_v^2 I$ respectively. Where :$\sigma_w^2 = \sigma_v^2 = e^{\alpha(1-mean\{\rho_i\})}$.

3.3 Algorithm steps

The KPF tracking algorithm based on kernel density estimation proposed in this article iterates according to the following steps:

Select target initial state $u_0^{(i)} = \{X_0^{(i)}, P_0^{(i)}, w_0^{(i)}\}_{i=1\ldots N}$ and initial observation value Z_0. Estimating the target template color distribution likelihood function $\hat{q} = \{q_j\}_{j=1\ldots m}$. Initial weight $w_0^{(i)} = 1/N$.

Obtain the position estimate according to equation (8): $\hat{X}_k = \{\hat{X}_k^{(i)}\}_{i=1\ldots N}$, and calculate the weight $\hat{w}_k^{(i)}$.

Perform mean translation for each particle: $\hat{X}_k^{(i)} = meanshift\{\hat{X}_k^{(i)}, \hat{w}_k^{(i)}\}_{i=1\ldots N}$, and correct the weight to:

$$\hat{w}_k^{(i)} = \sum_{j=1}^{m} \sqrt{\frac{q_j}{p_j(\hat{X}_k^{(i)})}} \delta[b(\hat{X}_k^{(i)}) - j].$$

Calculate the prediction variance $\hat{P}_k = \tilde{P}_{k-1} + R_k$, and calculate the Kalman gain: $K_k^{(i)} = \hat{P}_k^{(i)}(\hat{P}_k^{(i)} + R^{(i)})^{-1}$.

Update status:$\tilde{X}_k^{(i)} = \hat{X}_k^{(i)} + K_k^{(i)}(Z_k - \hat{X}_k^{(i)})$ and $\tilde{P}_k^{(i)} = \hat{P}_k^{(i)} - K_k^{(i)}\hat{P}_k^{(i)}$.

Estimate the target position according to equation (7).

Resample and reset the weight to $w_k^{(i)} = 1/N$. Jump to step 2.

4. EXPERIMENTAL RESULTS

The method described in this article was tested on a real image sequence, as shown in Figure 1. The image sequence size is 220×165 pixels, sampled at 30 frames/second, and the number of sampling particles is N=50. In all experiments, the initial state is determined by artificially specifying the outline of the target in the first frame of the image sequence. The red box indicates the locked target area.

It can be found from Figure 1 that when the aircraft rotates, emits light or changes in dimensions, the method introduced in this article can complete the tracking task well and has strong robust characteristics.

Figure. 1 Target tracking results of KPF algorithm based on kernel density estimation

5. SUMMARY

This paper starts from the kernel density estimation method and uses the similarity of the color distribution density function to achieve tracking of video image targets under the KPF filter architecture. And KPF was improved by simplifying the system model and mean shifting algorithm. Therefore, our proposed method requires fewer particles than existing KPF, improves tracking accuracy, and reduces computational cost. The effectiveness of this method was verified through experiments.

Yibin Vocational and Technical College project: ybzysc20-53 Video target recognition technology research.

Yibin Science and Technology Bureau Project: 2022SF002 Public safety research and application based on gait recognition.

6. REFERENCES

[1] Bao J, Zhang X, Zhang T, et al. Shadow-Enhanced Self-Attention and Anchor-Adaptive Network for Video SAR Moving Target Tracking. IEEE Transactions on Geoscience and Remote Sensing, 2023(6):1-13.
[2] Liu Wenya. The application of computer vision system on industrial robots . Computer fan, 2023 (17): 10-12.
[3] Zhanwei F. High Speed Moving Target Tracking Algorithm based on Mean Shift for Video Human Motion. Journal of Physic, 2021, 1744(4).
[4] Zhang J, Liu Z, Lin Y. Correlation Gaussian Particle Filter for Robust Visual Tracking. 2020 Chinese Control And Decision Conference (CCDC), Hefei, China, 2020: 4854-4857.
[5] C. Chang, R. Ansari. Kernel particle filter for visual tracking. IEEE Signal Processing Letters, 2005, 12(3): 242-245.
[6] Juseung C, Hoyong E, SeungMook B. A Wind Power Probabilistic Model Using the Reflection Method and Multi-Kernel Function Kernel Density Estimation. Energies, 2022, 15(24): 9436.
[7] Wang Y, Ding Y, Shahrampour S. TAKDE: Temporal Adaptive Kernel Density Estimator for Real-Time Dynamic Density Estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022,45(11):13831-13843,
[8] Sun, M. E. Davies, I. Proudler and J. R. Hopgood, Adaptive Kernel Kalman Filter. 2021 Sensor Signal Processing for Defence Conference (SSPD), Edinburgh, United Kingdom, 2021, 1-5.
[9] Tang Cundong, Wang Zhiping. Research and simulation of path estimation based on extended kalman filtering algorithm in video image. 2021 IEEE 4th International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 2021: 482-484.
[10] Cheng Xuecong, Liu Fucai, Huang Runan.. UWB indoor location algorithm based on Kalman filter and particle filter fusion. Acta Metrology, 2022, 43(10)1335-1340.
[11] Murata M , Kawano I, Inoue K. Degeneracy-Free Particle Filter: Ensemble Kalman Smoother Multiple Distribution Estimation Filter. IEEE Transactions on Automatic Control, 2022, 67(12): 6956-6961.

[12] Liu Ye, Jing XiaoYuan, Nie Jianhui, et al. Context-Aware Three-Dimensional Mean-Shift With Occlusion Handling for Robust Object Tracking in RGB-D Videos. IEEE Transactions on Multimedia, 2019, 21(3): 664-677.
[13] Yuan Q, Shi H, Xuan A T Y. Enhanced target tracking algorithm for autonomous driving based on visible and infrared image fusion . Journal of Intelligent and Connected Vehicles, 2023, 6(4): 237-249..
[14] Hui J, Yuan. Kalman filter, particle filter, and extended state observer for linear state estimation under perturbation (or noise) of MHTGR. Progress in nuclear engergy, 2022.
[15] Joseph A. Gallego-Mejia, Fabiuo A. González. DEMANDE: Density Matrix Neural Density Estimation,. IEEE Access, 2023(11): 53062-53078.