Research on Video Image Tracking Method Based on Kernel Density Estimation

Ya Zhang1, 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 $\{px_i\}_{i=1,\dots,n_{pix}}$, and assuming the centroid position is $Y = [x, y]^T$, then the color likelihood distribution in the candidate t The candidate target area is a rectangle, the coordinates of the pixel point
the centroid position is $Y = [x, y]^T$, then the color likelihood distribution
 $\hat{p}_j(X) = \frac{\sum_{i=1}^{n} |x_k(x)|^{\frac{Y - px_i}{h}}}{\sum_{i=1}^{n} |x_i(x)|^{\frac{Y - px_i}{h}}}\mathbf$ then the coordinates of the pixel points in the area are $\{px_i\}_{i=1...n_{pix}}$, and assuming
t, then the color likelihood distribution in the candidate target area is:
.....m (1) tangle, the coordinates of the pixel points in the area
 $y]^T$, then the color likelihood distribution in the cand

, j=1....m (1)

function, and the Epanechnikov kernel is selected he

Dirac function. 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 tar The candidate target area is a rectangle, the coordinates of the pixel points in

the centroid position is $Y = [x, y]^T$, then the color likelihood distribution in
 $\hat{p}_j(X) = \frac{\sum_{i=1}^{m} |x_k| \left(\frac{||Y - px_i||^2}{h}\right)^2}{\sum_{i=1}^{m} |x_i$ The candidate target area is a rectangle, the coordinates of the pixel points in the area are $\{px_i\}_{i=1,\ldots n_{pix}}$, and assuming the centroid position is $Y = [x, y]^T$, then the color likelihood distribution in the candidate t The candidate target area is a rectangle, the coordinates
the centroid position is $Y = [x, y]^T$, then the color like
 $\hat{p}_j(X) = \frac{\sum_{i=1}^{n} p_i k (\left\|\frac{Y - p x_i}{h}\right\|^2) \delta[b(p x_i) - j]}{\sum_{i=1}^{n} k (\left\|\frac{Y - p x_i}{h}\right\|^2)}$, j=1....m (1)
Among rectangle, the coordinates of the pixel points in the area are $\{px_i\}_{i=1,\dots n_{pix}}$, and assuming $[x, y]^T$, then the color likelihood distribution in the candidate target area is:
 $\frac{-1}{-1}$, $j=1,\dots m$ (1)
mel function, and t The candidate target area is a rectangle, the coordinates of the pixel points in the area are $\{px_i\}_{i=1...n_{p\downarrow x}}$, and
the centroid position is $Y = [x, y]^T$, then the color likelihood distribution in the candidate target a

The candidate target area is a rectangle, the coordi
the centroid position is
$$
Y = [x, y]^T
$$
, then the colo

$$
\hat{p}_j(X) = \frac{\sum_{i=1}^{n} x_i (\left\|\frac{Y - px_i}{h}\right\|^2) \delta[b(px_i) - j]}{\sum_{i=1}^{n} x_i (\left\|\frac{Y - px_i}{h}\right\|^2)}
$$
, j=1....m (1)
Among them: $k(\cdot)$ is the Kernel function, and the
value is set to 0.315; $\delta(\cdot)$ is the Dirac function.
In the first frame of the video, the target template
color likelihood distribution $\hat{q} = \{q_j\}_{j=1\ldots m}$ of the
The Dhetteabarrus coefficient of the two scalar bil

L_i E^{r κ}(|| ^π ||)

nong them: k(·) is the kernel function

lue is set to 0.315; δ(·) is the Dirac fu

the first frame of the video, the targe

lor likelihood distribution $\hat{q} = \{q_j\}_{j=1}$

e Bhattacharryya c ∧) = =1 $\n\hat{p}_j(X) = \frac{\sum_{i=1}^{n} p_i k(\left\|\frac{Y - \mu_{N,i}}{h}\right\|) \delta(b(px_i) - j)}{\sum_{i=1}^{n} p_i k(\left\|\frac{Y - \mu_{N,i}}{h}\right\|^{2})}$, j=1...m (1)

Among them: $k(\cdot)$ is the kernel function, and the Epanechnikov kernel is selected

value is set to 0.315; $\delta(\cdot$ In the first frame of the video, the target color likelihood distribution $\hat{q} = \{q_j\}_{j=1...m}$

The Bhattacharryya coefficients of the two $\rho(\hat{p}(X), \hat{q}) = \sum_{j=1}^{m} \sqrt{\hat{p}_j(X)\hat{q}_j}$ (2)

The similarity of color distribut annel of the video, the target template

od distribution $\hat{q} = \{q_j\}_{j=1...m}$ of the to

arryya coefficients of the two color like
 $\sum_{j=1}^{m} \sqrt{\hat{p}_j(X)\hat{q}_j}$ (2)

of color distribution is defined as:
 $\hat{p}(X), \hat{q}$ (3 value is set to 0.315; δ (\cdot) is the Dirac function.

In the first frame of the video, the target template can be determined color likelihood distribution $\hat{q} = \{q_j\}_{j=1,\dots,m}$ of the template can also

The Bhattachar

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

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

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

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

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

The similarity of color distribution is defined a
 $D = \sqrt{1 - \rho(\hat{p}(X), \hat{q})}$ (3)

The color likelihood function is:
 $p(Z^c|X) = \frac{1}{\sqrt{2\pi}\sigma_c} exp(-\frac{D^2}{2\sigma_c^2})$ (4)

Where: $\$ = $\sum_{j=1}^{m} \sqrt{\hat{p}_j(X)\hat{q}_j}$ (2)

y of color distribution is defined as:
 $(\hat{p}(X), \hat{q})$ (3)

elihood function is:
 $\frac{1}{2\pi\sigma_c} exp(-\frac{p^2}{2\sigma_c^2})$ (4)

is the variance of the color Gaussian distribution

ift

o the form harryya coefficients of the two color likelihood distributions are:
 $=\sum_{j=1}^{m} \sqrt{\hat{p}_j(X)\hat{q}_j}$ (2)

ty of color distribution is defined as:
 $\frac{\hat{p}_j(X)\hat{q}_j}{(\hat{p}(X),\hat{q})}$ (3)

selihood function is:
 $\frac{1}{2\pi\sigma_c} exp(-\frac$ $\rho(\stackrel{\wedge}{p}(X),\stackrel{\wedge}{q}) = \sum_{j=1}^{m} \sqrt{\stackrel{\wedge}{p}_j(X)\stackrel{\wedge}{q}_j}$ (2)

The similarity of color distribution is defined as:
 $D = \sqrt{1 - \rho(\stackrel{\wedge}{p}(X),\stackrel{\wedge}{q})}$ (3)

The color likelihood function is:
 $p(Z^c|X) = \frac{1}{\sqrt{2\pi}\sigma_c} exp(-\frac{D$ $\rho(\hat{p}(X), \hat{q}) = \sum_{j=1}^{m} \sqrt{\hat{p}_j(X)\hat{q}_j}$ (2)

The similarity of color distribution is defined as:
 $D = \sqrt{1 - \rho(\hat{p}(X), \hat{q})}$ (3)

The color likelihood function is:
 $p(Z^c|X) = \frac{1}{\sqrt{2\pi\sigma_c}} exp(-\frac{p^2}{2\sigma_c^2})$ (4)

Where: The similarity of color distribution is defined as:
 $D = \sqrt{1 - \rho(\hat{p}(X), \hat{q})}$ (3)

The color likelihood function is:
 $p(Z^c|X) = \frac{1}{\sqrt{2\pi\sigma_c}} exp(-\frac{D^2}{2\sigma_c^2})$ (4)

Where: σ_c is the variance of the color Gaussian distr The similarity of color distribution is defined as:
 $D = \sqrt{1 - \rho(\hat{p}(X), \hat{q})}$ (3)

The color likelihood function is:
 $p(Z^c|X) = \frac{1}{\sqrt{2\pi\sigma_c}} exp(-\frac{p^2}{2\sigma_c^2})$ (4)

Where: σ_c is the variance of the color Gaussian distr $D = \sqrt{1 - \rho(\hat{p}(X), \hat{q})}$ (3)

The color likelihood function is:
 $p(Z^c|X) = \frac{1}{\sqrt{2\pi}\sigma_c} exp(-\frac{\rho^2}{2\sigma_c^2})$ (4)

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

2.2 Mean shift

According to the formula (1 The color likelihood function is:
 $p(Z^c|X) = \frac{1}{\sqrt{2\pi\sigma_c}} exp(-\frac{D^2}{2\sigma_c^2})$ (4)

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

2.2 Mean shift

According to the formula (1), the color likelihood distri (4)
the color Gaussian distribution.
b, the color likelihood distribution in the ca
e gradient direction of the distribution to the
at the position of the center of mass of the pre-
e $\{px_i\}_{i=1,\dots,n_{pix}}$, then the steps of Where: σ_c is the variance of the color Gaussian distri

2.2 Mean shift

According to the formula (1), the color likelihood d

point can be moved along the gradient direction of the

distribution [5,12]. Assume that the to the formula (1), the color likelihood distribution in the candidate
be moved along the gradient direction of the distribution to the local
n [5,12]. Assume that the position of the center of mass of the predicte
lidate has of the predicted target is Y_0 and the pixel coord of the mean shift algorithm are:
ribution:
inpution:
THM BASED ON KERNEL DENSITY
IATION

of the candidate target area are
$$
\{px_i\}_{i=1\ldots n_p}
$$

\nSearch along the gradient direction of the col
\n
$$
Y_1 = \frac{\sum_{i=1}^{n_p} x_i' (\left\|\frac{px_i-Y_1}{h}\right\|^2) \cdot px_i \cdot w(px_i)}{\sum_{i=1}^{n_p} x_i' (\left\|\frac{px_i-Y_1}{h}\right\|^2) \cdot w(px_i)}
$$
\n
$$
W(px_i) \text{ is the weight:}
$$
\n
$$
w(px_i) = \sum_{j=1}^{m} \sqrt{\frac{q_j}{p_j(y_0)}} \delta[b(px_i) - j]. \qquad (6)
$$
\nIf $||Y_1 - Y_0|| < \varepsilon$, then end the loop; other
\n3. **KPF TARGET TRACKI**
\nKPF is a particle filter. The probability dist

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

ESTIMATION

distribution [5,12]. Assume that the position of the center of mass of the predicted target is Y_0 and the pixel coordinates

Search along the gradient direction of the color likelihood distribution:

Search along the **3. Solution 3.12**. Assume that the position of the center of mass of the predicted target is Y_0 and the pixel coordinates undidate target area are $\{px_i\}_{i=1,\ldots,n_{phys_i}}$, then the steps of the mean shift algorithm ar particles $u_k^{(i)} = \{x_k^{(i)}, w_k^{(i)}\}_{i=1}^k$ $\sum_{j=1}^{m} \sqrt{\frac{q_j}{p_j(Y_0)}} \delta[b(p x_i) - j]$. (6)
 $|Y_0|| < \varepsilon$, then end the loop; otherwise
 KPF TARGET TRACKING

article filter. The probability distribut
 $\sum_{k}^{(i)} = \{x_k^{(i)}, w_k^{(i)}\}_{i=1...N}$, where $x_k^{(i)}$

ing weight. In t $\{u^{(l)}_k, w^{(l)}_k\}_{i=1...N}$, where ((x_i)) = j]. (6)

(a) end the loop; otherwise $Y_0 = Y_1$, jump to step 1.

(GET TRACKING ALGORITHM BASED ON KERNEL DENSITY

ESTIMATION

The probability distribution of the state of the tracking object is approximated by **Y**
 $\sum_{k}^{(i)}$ is its
 \lim_{k} is its

the tracked

lytically in 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: **3.1 Particle morphology in KPF**
KPF follows the Kalman filter architecture, and
which are recorded as:
Prediction status: $\hat{X}_k = E[X_k | Z_k, ..., Z_1]$, correspo
Update status: $\tilde{X}_k = E[X_k | Z_{k-1}, ..., Z_1]$, correspor
Therefore, eac

∧ ∧ **J** and the second second the second sec

 $\widetilde{P}_k = [(X_k - \widetilde{X}_k)(X_k - \widetilde{X}_k)^T]$]

nology in KPF

alman filter architecture, and its paras:
 $k = E[X_k | Z_k, ..., Z_1]$, corresponding
 $= E[X_k | Z_{k-1}, ..., Z_1]$, corresponding v

ticle in KPF actually consists of three
 $\binom{r}{k}$ (7) **PF**
architecture, and its particle shape is divided into two stages:
...., Z₁], corresponding variance: $\hat{P}_k = [(X_k - \hat{X}_k)(X_k - \hat{X}_k)^T]$
..., Z₁], corresponding variance: $\tilde{P}_k = [(X_k - \hat{X}_k)(X_k - \hat{X}_k)^T]$
actually co is divided into two stages: prediction and update,
 $k = [(X_k - \hat{X}_k)(X_k - \hat{X}_k)^T]$
 $= [(X_k - \tilde{X}_k)(X_k - \tilde{X}_k)^T]$

ely $u_k^{(i)} = {X_k^{(i)}, P_k^{(i)}, w_k^{(i)}}_{i=1...N}$. The filtered output 3.1 Particle morphology in KPF

KPF follows the Kalman filter architecture, and its particle shape

which are recorded as:

Prediction status: $\hat{X}_k = E[X_k | Z_k, ..., Z_1]$, corresponding variance: \hat{P}_k

Update status: $\hat{X}_$ cture, and its particle shape is divided into two stages:

1], corresponding variance: $\hat{P}_k = [(X_k - \hat{X}_k)(X_k - \hat{X}_k)^T]$

], corresponding variance: $\tilde{P}_k = [(X_k - \tilde{X}_k)(X_k - \tilde{X}_k)^T]$

y consists of three parts, namely u is divided into two stages: prediction and
 \sum_{k} = [$(X_k - \hat{X}_k)(X_k - \hat{X}_k)^T$]
 \sum_{k} = [$(X_k - \tilde{X}_k)(X_k - \tilde{X}_k)^T$]

nely $u_k^{(i)} = {X}_k^{(i)}, P_k^{(i)}, w_k^{(i)}\}_{i=1...N}$. The filtere

[X_k, y_k]^T, X_k and y_k are the x-coordi to two stages: prediction and update,
 $\chi_k(X_k - \hat{X_k})^T$]
 $)(X_k - \tilde{X_k})^T$]
 $((\hat{U}, P_k^{(i)}, W_k^{(i)})_{i=1...N})$. The filtered output

and y_k are the x-coordinate and **3.1 Particle morphology in KPF**

KPF follows the Kalman filter architecture, and its particle shape is divided into two stages: predi

which are recorded as:

Prediction status: $\hat{X}_k = E[X_k | Z_k, ..., Z_1]$, corresponding varia Therefore, each particle in KPF actually consists of three parts, namely $u_k^{(i)} = \{X_k^{(i)}, P_k^{(i)}, w_k^{(i)}\}_{i=1,\dots,N}$. The filtered output ages: prediction and update,
 $\hat{X}_k \hat{Y}_k^T$]
 $\hat{X}_k^{(i)}$ _{$i=1...N$}. The filtered output **3.1 Particle morphology in KPF**
KPF follows the Kalman filter architecture
which are recorded as:
Prediction status: $\hat{X}_k = E[X_k | Z_k, ..., Z_1]$, co
Update status: $\tilde{X}_k = E[X_k | Z_{k-1}, ..., Z_1]$, co
Therefore, each particle in KPF

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

vided into two stages: prediction and
 $(X_k - \hat{X}_k)(X_k - \hat{X}_k)^T$]
 $X_k - \tilde{X}_k)(X_k - \tilde{X}_k)^T$]
 $(i) = {X_k^{(i)}, P_k^{(i)}, W_k^{(i)}\}_{i=1...N}$. The filtere
 X_k]^T, X_k and Y_k are the x-coordination movements, the random wall to two stages: prediction and upda
 χ_k)($X_k - \hat{X_k}$)^T]

)($X_k - \tilde{X_k}$)^T]

(i), $P_k^{(i)}$, $W_k^{(i)}$ }_{i=1...}, The filtered out

and y_k are the x-coordinate and movements, the random walk motion , follows the Kalman filter architecture

the are recorded as:

ction status: $\hat{X}_k = E[X_k | Z_k, ..., Z_1]$, contraditions $\hat{X}_k = E[X_k | Z_{k-1}, ..., Z_1]$, contraditions $\hat{X}_k = E[X_k | Z_{k-1}, ..., Z_1]$, contraditions of the status:
 $\hat{X}_k = \sum_{i$ **3.1 Particle morphology in KPF**

KPF follows the Kalman filter architecture, and its part

which are recorded as:

Prediction status: $\hat{X}_k = E[X_k | Z_k, ..., Z_1]$, corresponding v.

Update status: $\tilde{X}_k = E[X_k | Z_{k-1}, ..., Z_1]$, cor 3.1 Particle morphology in KPF

KPF follows the Kalman filter architecture, and its particle shape is divided into two stages: pre

which are recorded as:

Prediction status: $\hat{X}_k = E[X_k | Z_k, ..., Z_1]$, corresponding variance: $\left[(X_k - X_k)(X_k - X_k)^t \right]$
 $\left[(X_k - \tilde{X}_k)(X_k - \tilde{X}_k)^T \right]$
 $u_k^{(i)} = \{ X_k^{(i)}, P_k^{(i)}, w_k^{(i)} \}_{i=1...N}$. The
 $\left[X_k \right]$, $\left[X_k \right]$ and Y_k are the x-
 $\left[X_k \right]$, $\left[X_k \right]$ and Y_k are the x-
 $\left[X_k \right]$, $\left[X_k \right]$ and $\left[$ 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 d into two stages: prediction and update,
 $-\hat{X_k}(X_k - \hat{X_k})^T$]
 $-\tilde{X_k}(X_k - \tilde{X_k})^T$]
 $=\{X_k^{(i)}, P_k^{(i)}, W_k^{(i)}\}_{i=1...N}$. The filtered output
 $\hat{X_k}$ and Y_k are the x-coordinate and

mdom movements, the random walk mot **EXECT TO THE TARGE THE TARGE INTERED ASSES THE TARGE THE TARGE IN THE TARGE IN PROPERTION STAR THEORET SPECTIVELY.**
 Production status: $\hat{X}_k = E[X_k | Z_{k},..., Z_1]$, corresponding variance: $\hat{P}_k = [(X_k - \hat{X}_k)(X_k - \hat{X}_k)^T]$
 which are recorded as:

Which are recorded as:

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

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

$$
Z_k = X_{k-1} + \upsilon_k
$$

Therefore, each particle in KPF actually consi

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

3.2 Target motion model

At time k, the observation vector of the tar

y-coordinate of the target in the image respecti

m **E**[X_k] = $\sum_{i=1}^{N} w_k^{(i)} \tilde{X}_k^{(i)}$ (7)
 3.2 Target motion model

At time k, the observation vector of the tay-coordinate of the target in the image respecti

model is used to describe its motion state:
 $X_k = X_{k-1$ Therefore, each particle in KPF actually consists of
state is:
 $E[X_k] = \sum_{i=1}^{N} w_k^{(i)} \tilde{X}_k^{(i)}$ (7)

3.2 Target motion model

At time k, the observation vector of the target n

y-coordinate of the target in the image r del

ation vector of the target motic

get in the image respectively. Whe

be its motion state:

(8)

(8)
 $y_k]^T$, ω_k and v_k are process no:

Gaussian distribution. The corres
 y_k and v_k are process no:
 y_{k1} Among them: $X_k = [x_k, y_k]^T$, ω_k and v_k are process noise and observation noise respectively, which are independent of , and are process noise and observation noise respectively, which are independent of Example 1.1 The meant of the state is:

state is:
 $E[X_k] = \sum_{i=1}^{N} w_k^{(i)} \tilde{\chi}_k^{(i)}$ (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-coordina $^{2}_{w}I$ and $R = \sigma_{v}^{2}I$ respectively. State is.
 $E[X_k] = \sum_{i=1}^{N} w_k^{(i)} \tilde{X}_k^{(i)}$ (7)

3.2 Target motion model

At time k, the observation vector of the tay-coordinate of the target in the image respect

model is used to describe its motion state:
 $X_k = X_{k$ k, the observation vector of the target n
aate of the target in the image respectively.
used to describe its motion state:
 $x_1 + \omega_k$
(8)
nem: $X_k = [x_k, y_k]^T, \omega_k$ and v_k are proces
pr and obey Gaussian distribution. The co 2.1 $x_k = \sum_{i=1}^N w_k \, h_k$

3.2 Target motion model

At time k, the observation vector of the target mo

y-coordinate of the target in the image respectively. We

model is used to describe its motion state:
 $X_k = X_{k-1} + \omega_k$
 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 mo At time k, the observation vector of the targe

y-coordinate of the target in the image respective

model is used to describe its motion state:
 $X_k = X_{k-1} + \omega_k$
 $Z_k = X_{k-1} + \nu_k$ (8)

Among them: $X_k = [x_k, y_k]^T, \omega_k$ and v_k y-coordinate of the target in the image respectively. When
model is used to describe its motion state:
 $X_k = X_{k-1} + \omega_k$ (8)
 $Z_k = X_{k-1} + \nu_k$ (8)
Among them: $X_k = [x_k, y_k]^T$, ω_k and v_k are process noise
each other and obe U_k and U_k are process noise and observation distribution. The corresponding variant $tan(p_i)$.

(i) = { $X_0^{(i)}$, $P_0^{(i)}$, $W_0^{(i)}$ }_{i=1....}, and initial observation probably estimation probably and initial observati are process noise and observation

on. The corresponding variances

kernel density estimation propos
 $(w_0^{(i)})_{i=1...N}$ and initial observation
 $\{q_j\}_{j=1...m}$. Initial weight $w_0^{(i)} =$

equation (8): $\hat{X}_k = {\hat{X}_k^{(i)}}_{i$ tively. When the target makes random movements, the random walk motion
process noise and observation noise respectively, which are independent of
The corresponding variances are: $Q = \sigma_w^2 I$ and $R = \sigma_v^2 I$ respectively.
ne $X_k = X_{k-1} + \omega_k$ (8)
 $X_k = X_{k-1} + \omega_k$ (8)

Among them: $X_k = [x_k, y_k]^T$, ω_k and v_k are process noise and observation noise respectively, which are inder

acach other and obey Gaussian distribution. The corresponding vari observation noise respectively, which are independent of
g variances are: $Q = \sigma_w^2 I$ and $R = \sigma_v^2 I$ respectively.
tion proposed in this article iterates according to the
al observation value Z_0 . Estimating the target t Among them: $\lambda_k = [x_k, y_k]$, and b_k are process noise and observation noise respectively, which are mateproached better and obey Gaussian distribution. The corresponding variances are: $Q = \sigma_w^2 I$ and $R = \sigma_v^2 I$ respect Wh value only in this article iterates according to the
proposed in this article iterates according to the
proposed in this article iterates according to the
ervation value Z_0 . Estimating the target template
 $\frac{N}{N}$, an

 $(0, u^{(l)}), \ldots,$ estimation proposed in this article

ad initial observation value Z_0 . Estin

ial weight $w_0^{(i)} = 1/N$.
 $k = {\hat{X}_k^{(i)}}_{i=1...N}$, and calculate the wei

shif $t{\hat{X}_k^{(i)}, \hat{w}_k^{(i)}}_{i=1...N}$, and correct the nel density estimation proposed in this article iterates a
 $j_{k=1...N}^{(i)}$ and initial observation value Z_0 . Estimating the
 $j_{j=1...m}$. Initial weight $w_0^{(i)} = 1/N$.

ation (8): $\hat{X}_k = {\hat{X}_k^{(i)}}_{i=1...N}$, and calcula Calculate the prediction variance $\hat{P}_k = \tilde{P}_{k-1} + R_k$, and calculate the Kalman gain: $K_k^{(0)} = \hat{P}_{k}^{(0)} \hat{P}_{k}^{(0)} + \hat{P}_{k}^{(0)}$.

Color distribution likelihood function $\hat{q} = \{q_j\}_{j=1\ldots m}$. Initial weight $w_0^{(i)}$

 $\sum_{i=1}^N$ $\sum_{i=1}^N (i)$ k $l = 1...N$, and calcular $\hat{c}(i)$ k \cdot (i) Obtain the position estimate according to equation (8): $\hat{X}_k = \{\hat{X}_k^{(i)}\}_{i=1...N}$, and calculate th

Perform mean translation for each particle: $\hat{X}_k^{(i)} = meanshift\{\hat{X}_k^{(i)}, \hat{W}_k^{(i)}\}_{i=1...N}$, and corre
 $\hat{W}_k^{(i)} = \sum_{j$ 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 $\hat{w}_k^{(i)}$.

 \int_{-1}^{1} (i) \int_{-1}^{1} k \cdots \cdots \cdots \cdots $\int_{-1}^{1} (i) \int_{0}^{1} (i)$ $k \sim k$) $l = 1...N$, which (i) $\mathbf{u}^{(i)}$, \mathbf{v} and corr k $J = 1...N$, which seems

$$
\hat{w}_{k}^{(i)} = \sum_{j=1}^{m} \sqrt{\frac{q_{j}}{p_{j}(\hat{X}_{k}^{(i)})}} \delta[b(\hat{X}_{k}^{(i)}) - j]
$$

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

orrect the weight to:
 $(v_k^{(i)} = P_k^{(i)} (\hat{P}_k^{(i)} + R^{(i)})^{-1})$. $(k \choose k} (\stackrel{\wedge}{P_k}^{(i)} + R^{(i)})^{-1}.$ k , \cdots , \cdots t $\hat{w}_k^{(i)}$.

eight to:
 $(k + R^{(i)})^{-1}$. Perform mean translation for each particle: $\hat{X}_k^{(i)} = \text{meanshift} \{ \hat{X}_k^{(i)}, \hat{W}_k^{(i)} \}_{i=1...N}$, and correct the
 $\hat{W}_k^{(i)} = \sum_{j=1}^m \left[\frac{q_j}{p_j(\hat{X}_k^{(i)})} \delta[b(\hat{X}_k^{(i)}) - j] \right]$.

Calculate the prediction variance $\hat{P}_k = \$

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

article: $\hat{X}_k^{(i)} = \text{meanshift}\{ \hat{X}_k^{(i)}, \hat{W}_k^{(i)} \}_{i=1...N}$, and correct the weight
 $\hat{X}_k = \tilde{P}_{k-1} + R_k$, and calculate the Kalman gain: $K_k^{(i)} = \hat{P}_k^{(i)} (\hat{P}_k^{(i)})$
 $-\hat{X}_k^{(i)}$ and $\tilde{P}_k^{(i)} = \hat{P}_k^{(i)} - K_k^{(i)} \hat{P}_$ icle: $\hat{X}_k^{(i)} = \text{meanshift} \{ \hat{X}_k^{(i)}, \hat{W}_k^{(i)} \}_{i=1...N}$, and correct the weight to:

icle: $\hat{X}_k^{(i)} = \text{meanshift} \{ \hat{X}_k^{(i)}, \hat{W}_k^{(i)} \}_{i=1...N}$, and correct the weight to:
 $\hat{P}_{k-1} + R_k$, and calculate the Kalman gain: $K_k^{$ $w_k^{(s)} = \sum_{j=1}^m \sqrt{\frac{2}{p_j(X_k^{(s)}}} \delta[b(X_k^{(s)}) - j]$.
Calculate the prediction variance $\hat{P}_k = \tilde{P}_{k-1} + R_k$, and calculate the Kalman gain: $K_k^{(i)} = \hat{P}_k^{(i)} (P_k^{(i)} + R^{(i)})^{-1}$.
Update status: $\tilde{X}_k^{(i)} = \hat{X}_k^{(i)} + K_k^{(i)} (Z$ Calculate the prediction variance $\hat{P}_k = \tilde{P}_{k-1} + R_k$, and calculate the Kalman gain: $K_k^{(i)} = \hat{P}_k^{(i)}(\hat{P}_k^{(i)} + R^{(0)})^{-1}$.

Update status: $\tilde{X}_k^{(i)} = \tilde{X}_k^{(i)} + K_k^{(i)}(Z_k - \tilde{X}_k^{(i)})$ and $\tilde{P}_k^{(i)} = \hat{P}_k^{(i)} -$ 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 $\tilde{P}_k^{(i)} = \hat{P}_k^{(i)} - K_k$ Calculate the prediction variance $P_k = P_{k-1} + R_k$, and calculate the Kalr

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

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**

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.