PAPER *Special Section on Exploring Drone for Mobile Sensing, Coverage and Communications: Theory and Applications*

Adaptive Multi-Scale Tracking Target Algorithm through Drone

Qiusheng HE[†*], **Xiuyan SHAO**[††*], ***Nonmembers***, **Wei CHEN**[†††,††††*a)], ***Member***, **Xiaoyun LI**[†], **Xiao YANG**[†††], ***and*** **Tongfeng SUN**[†††], ***Nonmembers***

SUMMARY In order to solve the influence of scale change on target tracking using the drone, a multi-scale target tracking algorithm is proposed which based on the color feature tracking algorithm. The algorithm realized adaptive scale tracking by training position and scale correlation filters. It can first obtain the target center position of next frame by computing the maximum of the response, where the position correlation filter is learned by the least squares classifier and the dimensionality reduction for color features is analyzed by principal component analysis. The scale correlation filter is obtained by color characteristics at 33 rectangular areas which is set by the scale factor around the central location and is reduced dimensions by orthogonal triangle decomposition. Finally, the location and size of the target are updated by the maximum of the response. By testing 13 challenging video sequences taken by the drone, the results show that the algorithm has adaptability to the changes in the target scale and its robustness along with many other performance indicators are both better than the most state-of-the-art methods in illumination Variation, fast motion, motion blur and other complex situations.

key words: *target tracking, color feature, principal component analysis, scale adaptation*

1. Introduction

With the rapid development of drone technology, the drone has been widely used in many fields. Reference [1] uses a drone combined with a thermal far-infrared (FIR) camera to detect potential sinkholes over a large area. The drone is also used to make the strategy to prevent the potential possibility of the terrorism attack on the civilian areas [2]. Reference [3] describes a demonstrator application that uses the drone to monitor and detect the position and state of the person. A novel approach is presented to automatically determine the locations for soil samples based on a soil map created from drone imaging [4]. A drone is used to collect fast gas concentration data from underground coal fire [5]. Reference [6] introduces an approach able to predict copper accumulation points, using a combination of aerial photos, taken by drones. However, visual target tracking using the drone is not involved. The drone visual target tracking is a challenging task and often becomes very difficult due to illumination and scale change, occlusion, messy background, and fast motion and motion blur.

Tracking algorithm includes from early Mean Shift algorithm [9]–[11], particle filter algorithm [12]–[14], support vector machine (SVM) [15]–[17] to multiple instance learning algorithm [18]–[20]. Then the speed of correlation filter algorithm in the tracking process is paid more attention. Correlation filtering was first proposed for target tracking by Bolme et al [21], the simple grayscale feature training filter was used in the design of minimum output sum of squared error(MOSSE). João F. Henriques et al. [22] put forward CSK algorithm. KCF algorithm proposed in reference [23] applies HOG features to tracking algorithm, which improved the tracking precision. Luca Bertinetto et al. [24] proposed Staple algorithm, which integrated color histogram feature and HOG feature with a certain fusion-factor, and then after-fusion feature was used to train the filter, so the tracking precision was improved, but the tracking speed was greatly reduced. Hamed Kiani Galoogahi et al. [25] proposed the BACF algorithm, which took the background information into account in the tracking process, solved the problem of object occlusion and fast motion, and improved the tracking precision. In references [26]–[29], depth features were applied to the tracking algorithm. Although depth feature improved the tracking precision to a certain extent, the tracking speed was greatly reduced.

The tracking algorithms above do not solve the problem of target scale change. If the target shrinks, the filter will learn a lot of background information. If the target expands, the filter will be affected by the local texture of the target. Both situations are likely to produce unexpected results, leading to tracking drift and failure of tracking [30]–[32]. Martin Danelljan et al. [33] first applied CN(Color Name) to target tracking, used PCA to reduce the dimension of 11-dimensional features, and then the CSK tracking algorithm was used for tracking. The tracking precision was relatively high, but the effect was not ideal in the case of target scale change, partial occlusion and deformation. In the

Manuscript received November 23, 2018.
Manuscript revised February 2, 2019.
Manuscript publicized April 26, 2019.

[†]The authors are with School of Electronic Information Engineering, Taiyuan University of Science and Technology, Taiyuan 030024, P.R. China.

[††]The author is with Oulu Advanced Research on Service and Information Systems, University of Oulu, Oulu, FI-90540, Finland.

[†††]The authors are with Mine Digitization Engineering Research Center of the Ministry of Education, School of Computer Science and Technology, China University of Mining and Technology, Jiangsu, 221116, P.R. China.

[††††]The author is with College of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an, 710054, China.

[*]These authors contributed equally to this work and should be considered co-first authors.

a) E-mail: chenwdavior@163.com (Corresponding author)

DOI: 10.1587/transcom.2018DRP0040

paper, a multi-scale target tracking algorithm based on color attributes is proposed to solve the problem of scale change and achieve multi-scale target tracking through drone based on adaptive color feature.

2. Color Vision Tracking

Color features have rich expression and high identification. The CN algorithm maps the original RGB color space to 11 dimensional color attribute space [31], which is robust to the problem that the target is vulnerable to environmental changes in the process of the target tracking. The algorithm integrates the grey feature and color feature, and preprocessed each feature channel through the Hann window, finally obtains the comprehensive feature representation of grey level and color [32].

In order to shorten computing time, it adopts the dimensionality reduction technology of principal component analysis (PCA) to reduce the 11-dimensional color space to 2-dimensional color space. Let D_1 and D_2 represent the color space of 11-dimensional color space and 2-dimensional color space respectively, x is the color feature of D_1 dimension, and reduces the dimension by searching a mapping matrix with orthogonal column vectors of $D_1 \rightarrow D_2$. $\tilde{x}$ is obtained by linear mapping $\tilde{x} = B_p^T x$, which is the color feature of D_2 dimension.

After finish color feature extraction, then process the feature to track object [34]. CN algorithm is a discriminant tracking method, which determines the position of a new frame target according to the maximum responses of classifier. Therefore, the design of classifier is the main problem of tracking.

The color feature classifier adopts RLS classifier, which is a recursive least square algorithm. The CN algorithm minimizes a linear regularization function directly in the Reproducing Kernel Hilbert Space that is defined by the kernel. The classifier is obtained by training the image block with size $M \times N$ around the target, training sample $\tilde{x}_{m,n}$ is obtained by cyclic shift and $m \in \{0, \cdots, M-1\}$, $n \in \{0, \cdots, N-1\}$. The classifier is trained by minimizing regularized risk functional, and the risk functional is shown in Eq. (1).

$$\varepsilon = \sum_{j=1}^{p} \beta_j \left(\sum_{m,n} \left| \left\langle \varnothing \left(\tilde{x}_{m,n}^{j} \right), w^j \right\rangle - h^j(m,n) \right|^2 + \lambda \left\langle w^j, w^j \right\rangle \right) \tag{1}$$

where $\phi(\cdot)$ represents the function of image blocks map to the Hilbert space by the kernel function k, where the kernel uses the Gaussian kernel. $\tilde{x}_{m,n}^{j}$ is the target sample of the jth frame, h^j is the expected Gauss function output of $\tilde{x}_{m,n}^{j}$, and $\lambda \geq 0$ is the regularization parameter. This risk functional considers all frame errors, and β is the weighting coefficient of each frame error.

The solution of Eq. (1) can be expressed as the linear combination of inputs.

$$w^j = \sum_{m,n} a(m,n) \varnothing \left(\tilde{x}_{m,n}^{j} \right) \tag{2}$$

When the cost function is minimized, it should be satisfied.

$$A^P = \frac{A_N^P}{A_D^P} = F\{a\} = \frac{Y^P}{U_x^P + \lambda} = \frac{\sum_{j=1}^{p} \beta_j H^j U_x^j}{\sum_{j=1}^{p} \beta_j U_x^j \left(U_x^j + \lambda \right)} \tag{3}$$

where A_N^P, A_D^P, A^P, U_x^P, U_x^j, Y^P and H^j represent the corresponding Fourier transform respectively, $U_x^j = F\left\{u_x^j\right\}$, $F\{\cdot\}$ represents Fourier transforms, $u_x^j(m,n) = k\left(x_{xm,n}^{j}, x^j\right)$, the target model is updated according to Eq. (4).

$$\begin{cases} A_N^P = & (1-\beta) A_N^{P-1} + \beta Y^P U_x^P \\ A_D^P = & (1-\beta) A_D^{P-1} + \beta U_x^P \left(U_x^P + \lambda \right) \\ \hat{x}^p = & (1-\beta) \hat{x}^{p-1} + \beta \tilde{x}^p \end{cases} \tag{4}$$

where $\hat{x}^p$ is used to represent the target feature of the pth frame, which is estimated after updating. The mechanism makes it unnecessary to store redundant information in updating models, only needs to store the information $\left\{A_N^p, A_D^p, \hat{x}^p\right\}$ of the previous frame to ensure the tracking speed.

The output response of the classifier is $\hat{y}^p = F^{-1}\left(A^p U_z^p\right)$, where $U_z^p = F\left(u_z^p\right)$, $u_z^p(m,n) = k\left(z_{m,n}^{p}, \hat{x}^p\right)$, $z_{m,n}^{p}$ is the target feature that is extracted from the frame p, and $\hat{x}^p$ is target feature estimated of the pth frame after learning of classifier. Calculating the output response $\hat{y}$ and determine the location of the target in the next frame through its maximum value.

3. Modified CN Tracking Algorithm

The original CN algorithm was improved on the CSK algorithm, but it did not solve the problem of scale change. A scale estimation method based on MOSSE filter [35], [36]. Here the scale estimation strategy is added under the framework of CN tracking algorithm to achieve adaptive scale target tracking in the paper.

3.1 Scale Estimation Strategy

In the tracking process, the scale of target changes frequently due to motion. If the algorithm cannot adapt to the scale change of target, the output of the classifier will be affected to some extent, which causes tracking ineffective. The method adopted in the paper is to estimate the target location first, then estimate the scale based on the information of estimated location area, and finally update the results of target tracking according to the results of scale estimation. Scale estimation detects the change of target scale through correlation filter, so that the search area can be reasonably limited. Feature extraction process of scale estimation is shown in Fig. 1. Firstly, a series of rectangular regions with variable size are identified around the target, and the color

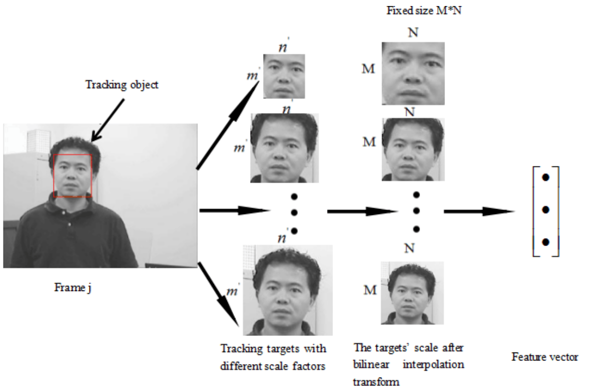
Fixed size M*N
Tracking object
n'
m'
N
M
Frame j
Tracking targets with different scale factors
The targets' scale after bilinear interpolation transform
Feature vector

Fig. 1 Feature extraction process of scale estimation.

features of each region are calculated. Then, the extracted features are used to train the RLS classifier, and the scale correlation filter is obtained. Supposes the target size in the current frame is $M \times N$, and the size of scale correlation filter is $S \times 1$. The areas with the size of $m' \times n'$ around the target are extracted successively, and the obtained sample is x^i_{scale}, $i = 1, \cdots, 33$, $\tau \in -(S-1)/2, \cdots, (S-1)/2$, $m' = \theta^\tau M$, $n' = \theta^\tau N$, θ is scale factor.

3.2 Rapid Scale Tracking

Taking into account the problem of the large amount of calculation during feature extraction, the dimension of 33 feature matrices that is obtained in Sect. 3.1 is unified into the dimension of the initial target region by bilinear interpolation in the paper. Then the 33 feature matrices after bilinear interpolation are integrated and the scale feature matrix x_{scale} of target is obtained. In addition, orthogonal trigonometric decomposition is used to reduce the dimension of the feature and reduce the amount of calculation.

1) Dimension reduction

In the tracking process, the tracking speed of the tracker is inversely proportional to the dimension of the feature. In the paper, considering the rapidity of calculation, orthogonal triangular decomposition is used to reduce the dimension of the feature. Through constructing a projection matrix $B_{p,scale}$ multiplying it by the target feature x_{scale} after bilinear interpolation transformation, the target feature in low dimensional space can be obtained. The dimension of the projection matrix $B_{p,scale}$ is $\tilde{d} \times d$, where d represents the dimension of the target feature before dimension reduction, $\tilde{d}$ represents the dimension of the target feature after dimension reduction and p represents the pth frame. Scale correlation filter is trained by minimum regularization risk functional, which is shown in Eq. (5).

$$\eta = \beta_p \sum_{m',n'} \left\| \hat{x}^p_{scale}(m',n') - B_{p,scale}\hat{x}^p_{scale}(m',n')\, B^T_{p,scale} \right\|^2 \tag{5}$$

To minimize the risk functional, let $B_{p,scale}B^T_{p,scale} = I$, the projection matrix can be solved through eigenvalue decomposition of autocorrelation matrix. Autocorrelation matrix is $Q_p = \sum \hat{x}^p_{scale}(m',n')\, \hat{x}^p_{scale}(m',n')^T$, eigenvalue decomposition formula is $Q_p = B_{p,scale}\Lambda_p B^T_{p,scale}$. Each row of the projection matrix $B_{p,scale}$ represents the eigenvector corresponding to the eigenvalue in Λ_p.

2) Dimension reduction scale tracking

By lessening the dimensionality, the tracking speed will be greatly improved without affecting the tracking precision. When the projection matrix is used to reduce the dimension of the feature, two projection matrices $B^x_{p,scale}$ and $B^u_{p,scale}$ are calculated for the target feature x^p_{scale} extracted and the target feature $u^p_{z,scale}$ that is estimated by learning of classifier respectively, the target feature extracted after reducing the dimension is $\tilde{x}^p_{scale} = B^x_{p,scale}x^p_{scale}$, and the target feature estimated by learning of classifier after reducing the dimension is $\tilde{u}^p_{z,scale} = B^u_{p,scale}u^p_{z,scale}$. In the tracking process, in order to improve calculation efficiency, the autocorrelation matrix is not constructed explicitly, but the projection matrix is obtained by QR decomposition of x^p_{scale} and $u^p_{z,scale}$ respectively.

Response $\hat{y}^p_{scale}$ of the detection scale filter can be obtained from Eq. (6), and the maximum scale of $\hat{y}^p_{scale}$ can be found as the scale of the target in the new frame.

$$\hat{y}^p_{scale} = F^{-1}\left(A^p G^p_{z,scale}\right) \tag{6}$$

where $G^p_z = F\{g^p_z\}$, $g^p_z(m,n) = k\left(B^x_{p,scale}z^p_{m,n}, B^u_{p,scale}\hat{x}^p_{scale}\right)$, $z^z_{m,n}$ represents the target feature extracted from the pth frame, and $\hat{x}^p_{scale}$ represents the target feature of the pth frame which is updated by the classifier through learning. Output response $\hat{y}^p_{scale}$ is calculated and the position of target in the next frame is determined by its maximum value. Then the target model is updated, and Eq. (7) is used for updating.

$$\begin{cases} \hat{A}^p_{scale} = (1-\beta)\hat{A}^{p-1}_{scale} + \beta A^p_{scale} \\ \hat{x}^p_{scale} = (1-\beta)\hat{A}^{p-1}_{scale} + \beta B^x_{p,scale}x^p_{scale} \end{cases} \tag{7}$$

where β represents scale learning factor, $\hat{A}^p_{scale}$ and $\hat{A}^{p-1}_{scale}$ represent the coefficient matrix of the current frame and the coefficient matrix after updating the previous frame respectively. $\hat{x}^p_{scale}$ and $\hat{x}^{p-1}_{scale}$ represent the target feature of the current frame and the target feature after updating the previous frame respectively. $B^x_{p,scale}$ is a projection matrix and may be got when reducing dimension. In target tracking, because the scale change of the target in two adjacent frames is very small, the position kernel correlation filter is used to detect the position of the target, then samples are collected around the target, and the scale of the target is detected by using the scale kernel correlation filter. In this way, the detection of target position and scale are completed. In the tracking algorithm, Gaussian function is used to output the position filter and scale filter, and the multi-channel color feature and gray value are selected for the target feature. The expected output y and y_s of the classifier are as shown in Eqs. (8) and (9), respectively.

$$y = \exp\left(-\left(\frac{p-p^*}{\sigma}\right)^2\right) \tag{8}$$

$$y_s = \exp\left(-\left(\frac{s-s^*}{\sigma_s}\right)^2\right) \tag{9}$$

where p represents the target position and p^* represents the center position of the target. s represents the total scale and s^* is the average value of all elements in s. σ and σ_{s} are the standard deviation of the scale kernel correlation filter and the position kernel correlation filter, respectively.

3.3 Arithmetic Flow

According to the above analysis, the proposed tracking algorithm includes mainly three parts. Firstly, the feature are preprocessed by Hann window and the target feature is extracted. Then we reduce the dimension of the target feature and calculate response value. Finally the target position and scale are updated. The detailed algorithm flow is as follows.

```
Input:
input image patch $I_t$.
the position of the previous frame $P_{t-1}$ and scale $S_{t-1}$.
position model $A^P_{t-1}$, $x^P_{t-1}$ and scale model $A^S_{t-1}$, $x^S_{t-1}$.
Output:
target location estimated $P_t$ and scale estimated $S_t$.
update position $A^P_t$, $x^P_t$ and update scale model $A^S_t$, $x^S_t$.
Position evaluation:
1,  According to the position of the previous frame of video, the color features are extracted in the current frame according to twice the target scale of the previous frame.
2,  use $Z$ and $A^P_{t-1}$, $x^P_{t-1}$ calculate $y$.
3,  calculate $\max(y)$, obtain target accurate position $P_t$.
Scale evaluation:
4,  Take current location of the target as the center, the color feature $Z'$ of 33 different scales was extracted.
5,  reduce the dimension to 17 dimension, then use $Z'$ and $A^S_{t-1}$, $x^S_{1-t}$ calculate $\hat{y}_S$.
6,  calculate $\max(\hat{y}_S)$, get accurate target scale $S_t$.
model updating:
7,  take sample $f_P$ and $f_S$.
8,  update position model $A^P_t$, $x^P_t$.
9,  update scale model $A^S_t$, $x^S_t$.
```

4. Experimental Evaluation

In order to verify tracking effect of the algorithm, 13 challenging videos in the OTB2015 dataset were selected for testing. The dataset provides accurate ground truth, the position and the size of initial frame to calculate tracking precision. The paper compared with 10 algorithms that are excellent tracking performance in recent years and include CSK, KCF, SAMF, SRDCF, DCF_CA, MOSSE_CA, STAPLE_CA and SAMF_CA and CN.

In the paper, the tracking effect of the algorithm is evaluated in terms of the accuracy of tracking and the change of scale. In order to facilitate the comparison, default parameters are used in the compared algorithms that are selected. The configuration of the experimental machine is as follows: Intel Pentium CPU G3250@3.20 GHz as CPU, the size of running memory is 2 GB, 64 bit operation system. The software environment: Win7 + Matlab2014.

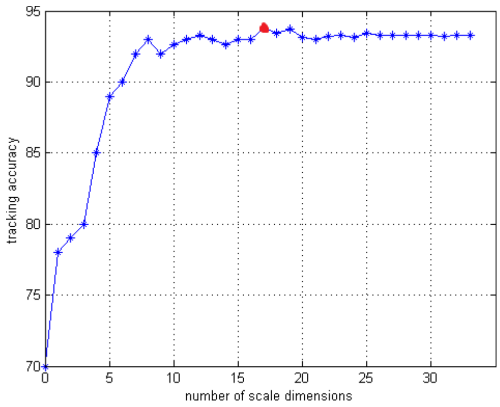

Fig. 2 The relationship between dimension of scale filter and tracking precision.

4.1 Fast Scale Estimation

In Sect. 3.2, dimension reduction is applied to the original scale estimation and the influence of dimensionality of scale correlation filter on tracking performance is analyzed in detail below.

As shown in Fig. 2, in the process of reducing dimension from 33 dimensions, tracking performance is basically consistent within a certain range. Results show that the dimension is set to 17, not only the tracking performance can be guaranteed, but also the dimension of the feature can be significantly reduced and the calculation speed can be improved.

4.2 Performance Analysis

Performance analysis is mainly divided into three parts: central position error, tracking success ratio and distance accuracy.

1) Central Position Error

The calculation formula of center position error is as follows.

$$CLE = \sqrt{||O - O_t||^2}$$

where O and O_t represent the real center coordinates of the target and the center coordinates that are obtained by the algorithm respectively. Center position error represents the error between the center that is obtained by algorithm and the real target center. The smaller the error is, the higher tracking precision is. The result of center position error is shown in Table 1. In general, the algorithm proposed in the paper has better performance in the central position error than other algorithms.

Table 1 Central position error results of visual tracking.

Video frame feature classification		CSK	KCF	SAMF	SRDCF	DCF_CA	MOSSE_CA	STAPLE_CA	SAMF_CA	CN	OUR
Changing in scale	Dog1	3.81	4.15	3.92	3.95	4.25	3.62	5.35	4.06	3.73	**3.43**
	Doll	44.7	7.58	2.96	2.62	5.44	29.5	3.86	3.46	7.25	**2.58**
	FleetFace	36.5	30.4	22.7	31.1	25.7	28.1	28.8	20	57.2	**16.8**
	Mhyang	3.61	3.64	2.42	2.09	3.61	3.58	3.02	3.02	3.57	**2.06**
Fast motion and motion blur	Jumping	86	29	28.7	4.47	34	**3.72**	6.14	38	49.3	4.01
	Deer	4.79	21.3	11.6	3.97	15.9	4.65	**3.94**	4.7	5.04	4.68
	Boy	20.1	2.62	4.05	**1.65**	2.4	2.42	2.56	2.42	2.61	2.16
	Fish	41.2	4.37	4.89	**3.42**	4.24	8.49	3.79	4.01	43.1	5.64
Light variation and background clutter	CarDark	3.23	5.26	2.5	1.57	4.82	2.64	**1.18**	1.76	3.59	1.86
	Dudek	9.87	9.38	8.73	12.6	11.3	15.6	13.6	9.78	15.7	**8.73**
	Shaking	17.2	109	62.2	85.4	7.63	72.4	148	**7.6**	27	10.5
	Skating1	7.78	7.71	**6.14**	20.3	7.38	79.9	6.28	102	7.64	6.96
	Trellis	18.8	7.73	2.72	**2.56**	7.48	68	3.05	2.56	21.2	3.03

Table 2 Tracking precision results.

Video frame feature classification		CSK	KCF	SAMF	SRDCF	DCF_CA	MOSSE_CA	STAPLE_CA	SAMF_CA	CN	OUR
Changing in scale	Dog1	100%	100%	99.9%	100%	100%	100%	99.9%	100%	100%	**100%**
	Doll	57.9%	97%	99.3%	99.3%	98.1%	76.4%	99.1%	99.2%	98.7%	**99.3%**
	Face	56.3%	60.50%	62.9%	59.7%	43.7%	38.2%	63.4%	**67.9%**	38.5%	62.2%
	Mhyang	100%	100%	100%	100%	100%	100%	100%	100%	100%	**100%**
Fast motion and motion blur	Jumping	5.11%	30.7%	27.2%	100%	69%	100%	97.1%	49.8%	18.5%	**100%**
	Deer	100%	81.7%	88.7%	100%	85.9%	100%	100%	100%	100%	**100%**
	Boy	84.4%	100%	100%	100%	100%	100%	100%	100%	100%	**100%**
	Fish	4.20%	100%	100%	100%	100%	100%	100%	100%	4.20%	**100%**
Light variation and background clutter	CarDark	100%	100%	100%	100%	100%	100%	100%	100%	100%	**100%**
	Dudek	84.3%	89.6%	**91.9%**	83.3%	87.1%	78.5%	81.8%	89.9%	75.1%	90.3%
	Shaking	56.4%	1.92%	2.74%	1.37%	94.5%	1.10%	3.29%	96.4%	20.8%	**96.6%**
	Skating1	98.8%	100%	100%	89.9%	100%	70%	100%	19.5%	100%	**100%**
	Trellis	81%	100%	100%	100%	100%	17.2%	100%	100%	65%	**100%**

Table 3 Track success rate.

Video frame feature classification		CSK	KCF	SAMF	SRDCF	DCF_CA	MOSSE_CA	STAPLE_CA	SAMF_CA	CN	OUR
Changing in scale	Dog1	65.3%	65.2%	82.6%	100%	64.9%	65.3%	100%	81.3%	65.3%	**100%**
	Doll	21.8%	64.4%	86.4%	99.7%	72.5%	57.1%	99.7%	81%	71.4%	**99.7%**
	Face	65.4%	63.5%	74.5%	66.3%	72.7%	61.4%	68.5%	79.3%	55.4%	**100%**
	Mhyang	100%	100%	100%	99.7%	100%	100%	99.5%	99.9%	100%	**100%**
Fast motion and motion blur	Jumping	4.79%	25.6%	24.3%	95.8%	68.1%	99.7%	70.6%	49.2%	6.39%	**99.8%**
	Deer	100%	81.7%	88.7%	100%	85.9%	100%	100%	100%	99%	**100%**
	Boy	84.2%	99.2%	99.3%	100%	99.3%	99.2%	100%	100%	99.2%	**100%**
	Fish	4.20%	100%	100%	100%	100%	100%	100%	100%	4.20%	**100%**
Light variation and background clutter	CarDark	99.2%	83.5%	100%	100%	92.9%	99.5%	100%	94.9%	94.7%	**100%**
	Dudek	87.3%	92.1%	100%	99.2%	97.5%	92.7%	79.1%	97.1%	91.5%	**100%**
	Shaking	58.1%	1.37%	1.37%	1.10%	92.1%	1.10%	3.01%	95.3%	21.1%	**96.4%**
	Skating1	36.8%	35.3%	80.3%	53.8%	35.3%	37.3%	62%	18.3%	35.3%	**89.5%**
	Trellis	59.1%	84%	99.3%	96.5%	84.2%	7.56%	99.3%	**100%**	56.4%	96.1%

2) Distance Precision
The formula for distance precision (DP) is $DP = m/n$, m is the number of video frames whose central position error is less than a certain threshold value, and n is the total number of frames of test data set. The threshold of the paper is 20 pixels. It can be seen from Table 2 that the algorithm in the paper has better tracking precision compared with other algorithms.
3) Tracking success ratio
The success ratio of tracking is defined as

$$OP = area\left(R_T \cap R_G\right) / area\left(R_T \cup R_G\right)$$

where the target region R_T is obtain by the algorithm, real target area R_G is marked by groundtruth, and $area\left(R_T \cap R_G\right)$ is the overlap area of two regions, and $area\left(R_T \cup R_G\right)$ is the area of the union of two regions. The higher the OP value is, the closer the region obtained by the algorithm is to the real target region, and the better the algorithm is. The higher the OP value is, the higher the accuracy rate is. And it shows the target is successfully tracked. Table 3 shows the results of tracking success ratio. It can be seen from the Table 3 that compared with other algorithms, the algorithm proposed in the paper has higher tracking success ratio.

4.3 Experimental Result

It can be seen from Tables 1–3 that the algorithm proposed in the paper is superior to other algorithms in tracking performance. In Figs. 3–5, some video frame screenshots of the algorithm in the tracking process will be given, which can more intuitively reflect the tracking effect.
1) Scale change
In the tracking process, scale change will affect the tracking precision. Video frames as shown in Fig. 3, Dog1 video changed significantly in scale in the tracking process. It can be seen from the first line in Fig. 1, before the 590th frame, all tracking algorithm were correct basically. However, in the 908th frame, the target scale became larger. It can be seen that only the proposed algorithm and STAPLE_CA, SRDCF could adapt to the changes in the scale, especially in the 1039th frame. In the subsequent frames, the scale of the target became smaller, and the algorithm in the paper also adapted to the change of scale and successfully tracked target. In the second line video Doll, the target of the 886th frame became larger, and most of the existing algorithms cannot adapt to the change of its scale. The 1637th frame and later, the CSK and MOSSE_CA have started to deviate from the target, failed to track. In the third line Fleetface video, the 614th frame, the target not only changed in scale, but also had a certain degree of rotation. The proposed algorithm showed good tracking effect. In the 663th frame

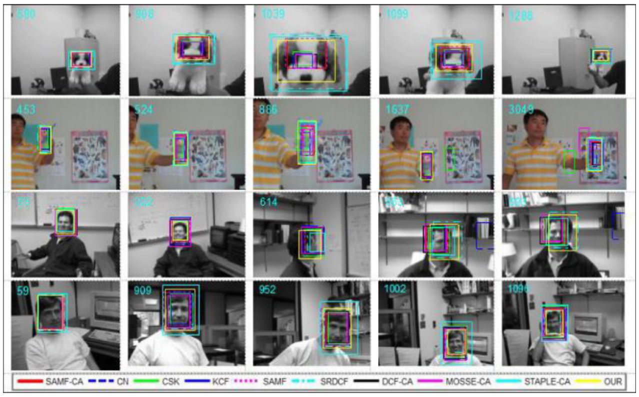

Fig. 3 Scale changes tracking results (Dog1, Doll, FleetFace, Mhyang).

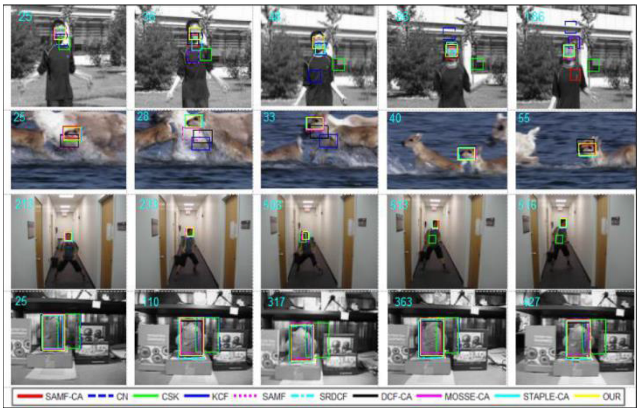

Fig. 4 Fast motion and motion blur tracking results (Jumping, Deer, Boy, Fish).

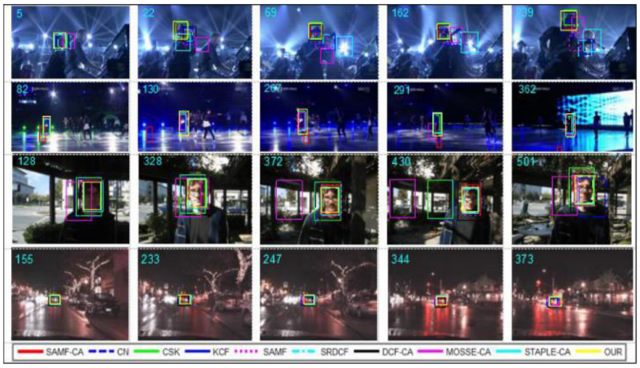

Fig. 5 Light variations and background clutter tracking results (Shaking, Skating1, Trellis, CarDark).

and 693th frame, CN has lost target. In the fourth line video Dudek, most of the algorithms were successful in tracking, but they were not able to adapt to the changes in the target scale, which affected the tracking success ratio. However, the proposed algorithm can still achieve 100% tracking success ratio.

2) Fast Motion and Motion Blur

In the process of tracking, fast moving speed of the target will make the target produce fuzzy pictures, which is a huge challenge for tracking. The Video shown in Fig. 4, the first line is screenshots of the tracking result of Jumping video. As the target range is small, the motion speed is fast, and the picture is blurred during the motion, many algorithms failed to track in the video frame. It can be seen from the figure that in 25th, 36th, 48th, 86th frame and the 186th frame, all the algorithms except SRDCF and the proposed algorithm cannot accurately track target. SRDCF deals with the scale in the algorithm, and add penalty terms in the algorithm, which can accurately track target. In the second line of Deer video, it can be seen from the 25th frame that DCF_CA and SAMG have deviated from the target. In the 28th frame, the KCF algorithm has also failed to track. Although tracked target successfully later, the tracking success ratio decreased. In the third line Boy1 video, the target not only has the characteristics of fast movement, but also has a certain degree of change in the scale. In the 508th frame, CSK started to deviate from the target, and the 513th frame has completely deviated. In the fourth line Fish video, most of the algorithms tracked accurately, but CSK and CN deviated from the target. In the group of videos, the tracking effect of the proposed algorithm was better than other algorithms, and it is the most obvious in Jumping video, achieving 99.8% success ratio.

3) Illumination Variation and messy background

Illumination variation and messy background are common in videos, and dealing with the disturbances in the tracking process is a major challenge. The first line shown in Fig. 5 is the results of tracking the moving objects, which varies greatly in illumination, and the target is similar to the background color, and the background is also very messy. Only in the 5th frame, MOSSE_CA has deviated from the target. In the 22th frame, only CSK, SAMF_CA and the proposed algorithm successfully can track target. In the 69th frame, 162th frame and 239th frame, DCF_CA also showed good tracking effect. In addition, other algorithms failed to track. In the second line Skating1 video, not only the illumination variation is obvious and the background is complex, but the target also has a certain degree of scale change. Because the algorithm cannot adapt the scale change, the tracking success ratio is not high.

In the 82th frame, SAMF_CA has already begun to deviate from the target, and the 30th frame has completely deviated from the target. In the 267th frame, the 291th frame and the 362th frame, the proposed algorithm can adapt its scale changes, and the tracking effect is better than other algorithms. In the third line Trellis video, MOSSE_CA started to deviate from the target in the 128th frame, MOSSE_CA, CSK, and CN all failed to track. In the fourth line CarDark video, most of the algorithms successfully tracked target. In the group of videos, the proposed algorithm achieved good tracking effect, especially in Shaking and Skating1 video, 96.4% of the tracking success ratio and 89.5% of the tracking ratio were better than other algorithms.

5. Conclusion

In the paper, the multi-scale target tracking algorithm based on adaptive color features through the drone is proposed, which map the original RGB image to the 11-dimensional color attribute feature space, so as to achieve the robust representation of color features. Nuclear tracking method based on color feature is adopted for tracking target quickly and accurately. In the scale estimation, the color feature is calculated through rectangular regions around the target, and the scale correlation filter is designed to solve the problem of target scale change in the tracking process. The advantages of the multi-scale tracking algorithm based on adaptive color features proposed in this paper are as follows:

(1) Strong adaptability for the changes in the target scale (such as the results of Dog and Doll video sequences).

(2) For fast motion and motion blur video, the algorithm in the paper is fast in operation and can successfully track with high accuracy.

(3) It has good resistance to illumination variation and messy background, because the original RGB color is mapped to CN feature space, which improves the robustness of color representation.

Experimental results show that the proposed algorithm applied in the drone in the paper not only maintains the advantages of fast and accurate color feature tracking, but also it achieves robust tracking under scale change, fast motion and background interference. Although the problem of tracking failure may occur in high-speed drone, the cloud computing method and edge calculation method for the high-speed vehicles [37]–[39] provide a theoretical basis for the application of the algorithm in the drone.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 51874300), the National Natural Science Foundation of China and Shanxi Provincial People's Government Jointly Funded Project of China for Coal Base and Low Carbon (Grant No. U1510115), the National Natural Science Foundation of China (Grant Nos. 51874299, 51104157), the Qing Lan Project and the China Postdoctoral Science Foundation (Grant No. 2013T60574).

References

[1] E.J. Lee, S.Y. Shin, C.K. Byoung, and C. Chang, "Early sinkhole detection using a drone-based thermal camera and image processing," Infrared Phys. Techn., vol.78, pp.223–232, Sept. 2016.

[2] H.W. Tae, "Anti-nuclear terrorism modeling using a flying robot as drone's behaviors by global positioning system (GPS), detector, and camera," Ann. Nucl. Energ., vol.118, pp.392–399, Aug. 2018.

[3] A.K. Singh, A. Swarup, A. Agarwal, and D. Singh, "Vision based rail track extraction and monitoring through drone imagery," ICT Express, In press, corrected proof, Available online 12, Dec. 2017.

[4] J. Huuskonen and T. Oksanen, "Soil sampling with drones and augmented reality in precision agriculture," Comput. Electron. Agric., vol.154, pp.25–35, Nov. 2018.

[5] L. Dunnington and M. Nakagawa, "Fast and safe gas detection from underground coal fire by drone fly over," Environ. Pollut., vol.229, pp.139–145, Oct. 2017.

[6] A. Capolupo, S. Pindozzi, C. Okello, N. Fiorentino, and L. Boccia, "Photogrammetry for environmental monitoring: The use of drones and hydrological models for detection of soil contaminated by copper," Sci. Total Environ., vol.514, pp.298–306, May 2015.

[7] H. Zhou, X. Wang, Z. Liu, S. Yamada, and Y. Ji, "Resource allocation for SVC streaming over cooperative vehicular networks," IEEE Trans. Veh. Technol., vol.67, no.9, pp.7924–7936, Sept. 2018.

[8] R. An, Z. Liu, H. Zhou, and Y. Ji, "Resource allocation and layer selection for scalable video streaming over highway vehicular networks," IEICE Trans. Fundamentals, vol.E99-A, no.11, pp.1909–1917, Nov. 2016.

[9] S.F. Razavi, H. Sajedi, and M.E. Shiri, "Integration of colour and uniform interlaced derivative patterns for object tracking," Image Process., vol.10, no.5, pp.381–390, 2016.

[10] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol.39. no.4, pp.640–651, 2017.

[11] L.M. Hu, L.L. Duan, X.D. Zhang, et al., "Moving object detection based on the fusion of color and depth information," J. Electron. Inform. Technol., vol.36, no.9, pp.2047–2052, 2014.

[12] S. Wang, R. Clark, H. Wen, and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks," 2017 IEEE International Conference on Robotics and Automation (ICRA), pp.2043–2050, Marina Bay, Singapore, May 2017.

[13] W. Song, J. Zhu, Y. Li, and C. Chen, "Image alignment by online robust PCA via stochastic gradient descent," IEEE Trans. Circuits Syst. Video Technol., vol.26, no.7, pp.1241–1250, July 2016.

[14] P. Ammirato, P. Poirson, E. Park, J. Kosecka, and A.C. Berg, "A dataset for developing and benchmarking active vision," 2017 IEEE International Conference on Robotics and Automation (ICRA), pp.1378–1385, Marina Bay, Singapore, May 2017.

[15] J. van de Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications," IEEE Trans. Image Process., vol.18, no.7, pp.1512–1523, 2009.

[16] F.S. Kuan, J. Van de Weijer, and M. Vanrell, "Modulating shape features by color attention for object recognition," Int. J. Comput. Vision, vol.98, no.1, pp.49–64, 2012.

[17] F.S. Kuan, R.M. Anwer, and J. Van de Weijer, "Color attributes for object detection," IEEE Conference on Computer Vision and Pattern Recognition, pp.3306–3313, Rhode Island, USA, June 2012.

[18] D. Martin, M. Anelljan, and F.S. Kuan, "Adaptive color attributes for real-time visual tracking," IEEE Conference on Computer Vision and Pattern Recognition, pp.1090–1097, Columbus, USA, June 2014.

[19] G. Costante, M. Mancini, P. Valigi, and T.A. Ciarfuglia, "Exploring representation learning with CNNs for frame-to-frame ego-motion estimation," IEEE Robot. Autom. Lett., vol.1, no.1, pp.18–25, 2016.

[20] J. Ning, L. Zhang, D. Zhang, and C. Wu, "Scale and orientation adaptive mean shift tracking," IET Comput. Vis., vol.6, no.1, pp.52–61, 2012.

[21] T. Vojir, J. Noskova, and J. Matas, "Robust scale-adaptive mean-shift for tracking," Pattern Recogn. Lett., vol.49, no.1, pp.250–258, 2014.

[22] Y. Wu, J. Lim, and M.H. Yang, "Online object tracking: A benchmark," IEEE Conference on Computer Vision and Pattern Recognition, pp.2411–2418, Oregon, USA, 2013.

[23] P. Richtarik and M. Takac., "Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function," Math. Program., vol.144, no.1, pp.1–38, April 2014.

[24] A. Mojsilovic, "A computational model for color naming and describing color composition of images," IEEE Trans. Image Process., vol.14, no.5, pp.690–699, 2005.

[25] J. van de Weijer, C. Schmid, and J. Verbeek, "Learning color names

for real-world applications," J. Software, vol.21, no.4, pp.586–596, 2010.

[26] R. Clark, S. Wang, H. Wen, A. Markham, and N. Trigoni, "VINet: Visual-inertial odometry as a sequence-to-sequence learning problem," arXiv e-prints, arXiv:1701.08376, Jan. 2017. https://ui.adsabs.harvard.edu/#abs/2017arXiv170108376C

[27] P. Meer, V. Ramesh, and D. Comaniciu, "Kernel-based object tracking," IEEE Trans. Pattern Anal. Mach. Intell., vol.25, no.5, pp.564–575, 2003.

[28] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan, "Locally orderless tracking," IEEE Conference on Computer Vision and Pattern Recognition, pp.1940–1947, Rhode Island, USA, 2012.

[29] J. Van de Weijer, C. Schmid, and J. Verbeek, "Learning color names from real-world images," IEEE Conference on Computer Vision and Pattern Recognition, pp.1–8, Minneapolis, Minnesota, USA, 2007.

[30] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," arXiv preprint, arXiv:1405.3531, 2014.

[31] R. Clark, S. Wang, A. Markham, N. Trigoni, and H. Wen, "VidLoc: 6-dof video-clip relocalization," arXiv preprint, arXiv:1702.06521, 2017.

[32] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," IEEE Conference on Computer Vision and Pattern Recognition, pp.142–149, Hilton Head, SC, USA, 2000.

[33] M. Danelljan, F.S. Khan, and M. Felsberg, "Adaptive color attributes for real-time visual tracking," J. Software, vol.21, no.4, pp.586–596, 2010.

[34] M. Jaderberg, K. Simonyan, A. Zisserman, et al., "Spatial transformer networks," Advances in Neural Information Processing Systems, pp.2017–2025, 2015.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proc. IEEE conference on computer vision and pattern recognition, pp.770–778, 2016.

[36] H. Possegger, T. Mauthner, and H. Bischof, "In defense of color-based model-free tracking," IEEE Conference on Computer Vision and Pattern Recognition, pp.2113–2120, Boston, USA, 2015.

[37] J. Feng, Z. Liu, C. Wu, and Y. Ji, "HVC: A hybrid cloud computing framework in vehicular environment," The 5th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, pp.9–16, San Francisco, USA, April 2017.

[38] J. Guo, Z. Song, Y. Cui, Z. Liu, and Y. Ji, "Energy-efficient resource allocation for multi-user mobile edge computing," IEEE Global Communications Conference (GLOBECOM), Singapore, pp.640–648, Dec. 2017.

[39] Z. Liu, M. Dong, B. Zhang, Y. Ji, and Y. Tanaka, "Real-time multi-view video streaming in highway vehicle ad-hoc networks (VANETs)," IEEE Global Communication Conference, Washington DC, DC, USA, Dec. 2016. DOI: 10.1109/GLOCOM.2016.7842230.

Qiusheng He received his B.S. and M.S. degrees from Taiyuan University of Technology in Taiyuan, China, in 1998 and 2004 respectively, and his Ph.D. degree from China University of Mining & Technology, Beijing, China, in 2007. He is currently an associate professor and a Master Supervisor. He has authored and co-authored over 50 conference and journal papers, presided over the completion of 5 national and provincial projects, and has been granted 4 national invention patents in her research areas. His research interests include nonlinear control theory research and application, machine learning, image processing.

Xiuyan Shao is a postdoctral researcher in Faculty of Information Technology and Electronic Engineering in University of Oulu, Finland. Her current research interest includes image processing, visual tracking, statistical signal processing and machine learning.

Wei Chen received the B.Eng. degree in medical imaging and the M.S. degree in paleontology and stratigraphy from China University of Mining and Technology, Xuzhou, China, in 2001 and 2005, respectively, and the Ph.D degree in communications and information systems from China University of Mining and Technology, Beijing, China, in 2008. In 2008, he joined the School of Computer Science and Technology, China University of Mining and Technology, where he is currently a professor. He is a member of IEEE, ACM and EAI. His research interests include machine learning, image processing, and computer networks.

Xiaoyun Li received her B.S. from Taiyuan University of Technology in 2018. She is currently a graduate student at Taiyuan University of Technology. Her research interests include machine learning, image processing.

Xiao Yang received her B.S. from North West Agriculture and Forestry University in 2018. She is currently a graduate student at China University of Mining and Technology. Her research interests include machine learning, image processing.

Tongfeng Sun received his B.S. degree in industrial automation from Northwestern Polytechnical University in 1999. He received his M.S. degree in computer application technology in 2004 and Ph.D. degree in detection technology and automation devices in 2012 from China University of Mining and Technology. He is currently an associate professor in the university. His research interests are in the areas of intelligent information processing, machine learning and pattern recognition.