Journal of Network Intelligence
Taiwan Ubiquitous Information

Volume **9**, Number **4**, November **2024**

Video Target Tracking Based on KPCA and Structured Support Vector Machines

Wei-Wei Ma*

School of Computer and Communication
Jiangsu Vocational College of Electronics and Information, Huai'an 223003, P. R. China
marqul2003@163.com

Yan Gao

School of Architecture, Decoration and Art Design
Jiangsu Vocational College of Electronics and Information, Huai'an 223003, P. R. China
57408943@qq.com

Da Huang

Sehan University, Jeollanam-do 58447, South Korea
57408943@qq.com

*Corresponding author: Wei-Wei Ma
Received January 9, 2024, revised April 30, 2024, accepted August 9, 2024.

Abstract. *Video target tracking is widely used in many fields such as human-computer interaction and augmented reality. Since video scenes contain many complex information such as lighting, occlusion, background interference, etc., and the motion changes of the target itself, it makes it a great challenge to study robust target tracking algorithms. Therefore, a specific study on video target tracking algorithms is done and a video target tracking method based on Kernel Principal Component Analysis (KPCA) and structured support vector machine is proposed. Firstly, HOG features and LBPHF features are computed separately in each sliding window, and the overlapping detection windows are fused using the Non-Maximum Suppression (NMS) method, so that the two are combined to form the joint features. Secondly, after extracting the joint HOG-LBPHF features of the video target, they are mapped by KPCA to the feature vectors that are more favourable for target classification. Then, based on the structured support vector machine, the inter-frame target matching fluctuation value is used to judge whether the target is abnormal or not, so as to decide whether to update the support vectors or not, and combined with Kalman filtering to predict the target position in the next frame and correct the tracking results. The experimental results show that the proposed method satisfies the requirements of high detection rate and low computational complexity. For four types of public video datasets, the proposed method has obvious advantages in four key performance indexes, such as tracking accuracy rate and tracking rate, and has strong adaptability in dealing with large-scale video target tracking.*
Keywords: Target tracking; structured support vector machine; feature extraction; kernel principal component analysis; Kalman filtering;

1. **Introduction.** Video target tracking technology, as an important branch of computer vision technology, has been widely used in many fields such as automatic driving [1, 2] and safe city [3, 4]. Target tracking is accomplished by acquiring image samples of video target data by frame, and performing image feature analysis through the context and internal

correlation of image data in adjacent frames, and then completing motion prediction [5] and localisation analysis [6] of the target.

In video target tracking research, feature extraction and target recognition are the core aspects of target tracking [7, 8], the former requires feature extraction of the calibrated image samples, while the latter needs to analyse and generalize the features to obtain the calibrated image recognition results. In recent studies, feature extraction is mainly done by using various dimensionality reduction methods for feature optimisation [9] to reduce data dimensionality and redundancy, and target recognition is mostly done by using support vector machines [10] and neural network algorithms [11, 12].

In the field of video target tracking, Structured Support Vector Machine (SSVM) [13] can be used to build a structured model of the target and improve the tracking effect by learning the constraints of the target. Kernel Principal Component Analysis (KPCA) [14], on the other hand, can be used to extract the nonlinear features of the target, which can better capture the changes of the target. Combining the two, the structural and nonlinear features of the target can be taken into account in tracking and optimised to improve the tracking effect. Therefore, in this paper, KPCA is used for image feature optimisation to obtain more valuable target image samples and SSVM is used for target classification, resulting in the KPCA-SSVM algorithm. The aim of this study is to use Structured Support Vector Machine for video target tracking and with the help of Kernel Principal Component Analysis for video target feature dimensionality reduction and de-redundancy place to enhance the video target classification adaptation.

1.1. **Related Work.** In both the fields of target detection and target tracking, the key issues are how to effectively describe the target and how to allow the computer to accurately recognise the target.

Collins et al. [15] proposed tracking target features based on the log-likelihood ratio of the foreground histogram to the background histogram. Bousetouane et al. [16] proposed a target tracking algorithm using colour and texture to describe the target features. Weiskopf et al. [17] proposed a tracking method based on texture features, which overcomes the effect of lighting variations on tracking in time-varying volumetric data. The segmentation and tracking results are used as inputs and outputs of each other, i.e., on the one hand, the segmentation results are provided as inputs to the tracking, while on the other hand, the information output from the tracking results is also inputted to the segmentation, so that the tracking can be formed in consecutive frames in a complementary way. Zhang et al. [18] proposed to classify tracking as a binary classification algorithm problem, whose main method is to achieve the tracking purpose by classifying video frames, and the formation of the classifier needs to be learnt from the previous frames. Li et al. [19] proposed a target tracking method based on online learning, which mainly introduces the Boosting algorithm to achieve online feature selection. Zhang et al. [20] proposed the use of Support Vector Machines to train classifiers, in terms of introducing SVMs into visual tracking, but the disadvantage of this method is that it cannot be updated online. In addition, it may lead to tracking drift phenomenon due to the limitation of the training target class. Zhang et al. [21] improved on this by proposing an SVM-based semi-supervised classification method to achieve target classification, which improves the classification ability by implementing updates through unlabelled samples. Li et al. [22] proposed to apply structured output SVM to target detection tracking.

KPCA [23] is a method that extends linear PCA to nonlinear situations. In the field of video target tracking, KPCA has better nonlinear modelling and higher recognition accuracy, and can better adapt to nonlinear data. Compared to traditional linear PCA, KPCA can describe more nonlinear features because it maps the feature space into a

high-dimensional or infinite-dimensional feature space, thus better adapting to nonlinear relationships in video target tracking.

1.2. **Motivation and contribution.** The traditional structured support vector machine can produce tracking drift phenomenon when target tracking if there are problems of target half-obscuration and motion crossover. In addition, the HOG feature [24] can capture the edge and local shape information of the video target well, so this feature is widely used in the field of video target tracking at this stage. However, HOG features still have defects, for example, if the background is cluttered, the apparent performance of HOG becomes poor.

Therefore, in order to solve the above problems, this work proposes a video target tracking method based on KPCA-SSVM. The main innovations and contributions of this work include:

(1) Feature extraction is improved to improve the recognition rate of the video target tracking system. HOG features and LBPHF features [25] are computed separately in each sliding window, and the overlapping detection windows are fused using the Non-Maximum Suppression (NMS) [26] method.

(2) After extracting the joint HOG-LBPHF features of the video target, it is mapped by KPCA to a feature vector that is more favourable for target classification.

(3) Aiming at the existing structured support vector machine tracking algorithms which cannot cope well with the shortcomings such as motion crossing and semi-obscuration, Tsai proposed an improved online target tracking method based on SSVM.

2. KPCA-based feature extraction and dimensionality reduction.

2.1. **HOG-LBPHF Feature Extraction.** Feature extraction, as a fundamental problem in the disciplines of computational vision and image processing, is mainly used to describe the image information and verify whether any pixel in the picture represents a feature. To obtain as much image information as possible in feature extraction, image preprocessing is needed to pre-process the image, such as basic image processing methods such as smoothing, sharpening, filtering, etc.; or more advanced processing such as image fusion, super-resolution, etc.

The HOG feature proposed in 2005 can capture the edge and local shape information of the video target very well, so this feature is widely used in the field of video target tracking at this stage. However, HOG features still have defects, for example, if the background is cluttered, the apparent performance of HOG becomes poor, but the Fourier transform-based LBP can compensate for the above defects. Therefore, this work improves the feature extraction to improve the recognition rate of the video target tracking system. Specifically, the LBPHF features are combined with HOG features. The HOG features and LBPHF features are computed separately in each sliding window, and the overlapping detection windows are fused using the NMS method, so that the two are combined to form a joint feature.

HOG features are feature descriptors based on local information statistics, which consist of histograms of local gradients of an image by computing and counting them. In an image, the orientation density distribution of gradients or edges can well describe the local target representation. The specific extraction method includes the following steps.

(1) Image normalisation.

The input colour image is firstly greyscaled and secondly the image is normalised. The normalisation uses Gamma correction to normalise the colour space of the image in order to reduce the effect of local shading and illumination variations in the image.The Gamma

correction compression method is shown below:

$$I(x,y) = I(x,y)^{\gamma} \tag{1}$$

(2) Gradient calculation.

The 1-D discrete gradient template is the simplest method of calculating the gradient, and the horizontal and vertical gradients are calculated by the gradient template as follows, respectively:

$$\begin{cases} G_h(x,y) = H(x+1,y) - H(x-1,y) \\ G_v(x,y) = H(x,y+1) - H(x,y-1) \end{cases} \tag{2}$$

where H is the image; G_h and G_v represent the horizontal and vertical gradients, and the convolution kernels computed for both are $[-1,0,1]$ and $[-1,0,1]^T$ respectively. The gradient at pixel point (x,y) is $G(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2}$ and the gradient direction is $\alpha(x,y) = \tan^{-1}\left(\frac{G_x(x,y)}{G_y(x,y)}\right)$

(3) Construction of orientation histograms.

A sliding window method is used, where the sliding window slides over the image to be tested from left to right and from top to bottom. The image is divided into a number of interval blocks within the sliding window, and the interval blocks are divided into a number of grid cells. A histogram of the gradient direction is constructed for each grid cell of the image. The sliding window, interval blocks, and grid cells are extracted as shown in Figure 1.

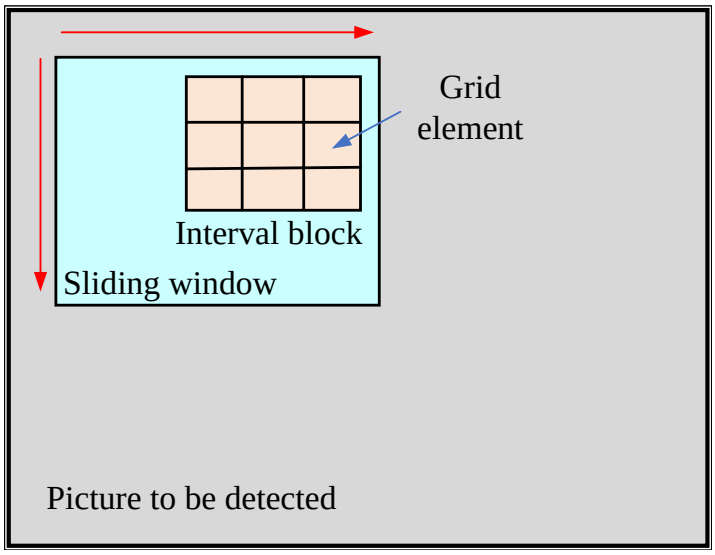

Figure 1. MMH-PSO algorithm for membrane structure

Each pixel point in a grid cell votes for an orientation-based histogram channel, with voting taking the form of a weighted ballot, with each weight calculated from the gradient magnitude of that pixel value. The histogram channel is an undirected gradient distributed evenly over 0-180°. The sliding window parameters were set to 128×64, with 3×3 grid cells per rectangular block and 6×6 pixels in each grid cell. The gradient information of the 6×6 pixels in the grid cells is counted using 9 histogram channels.

(4) HOG feature vector normalisation.

The histograms of the gradient directions of all cells in the interval block are normalised. The purpose is to make the feature vector space overcome the effects of light, shadow and

edge variations. Since L^1 normalisation reduces the computational effort to some extent on the basis of the same effect, L^1 is used in this paper to normalise the HOG.

$$v \leftarrow v/\left(\|v\|_1 + \epsilon\right) \tag{3}$$

LBP (Local Binary Pattern) operator [27] is an operator used to describe the local texture features of an image to solve the texture classification problem. LBP operator has a good classification effect and is computationally efficient, and it has been used in the field of text recognition with good results, and later used in face recognition as well as pedestrian detection, which has also achieved good results. Nowadays, LBP is widely used as an efficient descriptive algorithm in the fields of target detection and face recognition.

The definition of the original LBP operator is shown below:

$$\mathrm{LBP} = \sum_{i=0}^{P-1} u(g_i - g_c)2^i \tag{4}$$

$$u(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{5}$$

The original LBP operator extraction method is shown in Figure 2.

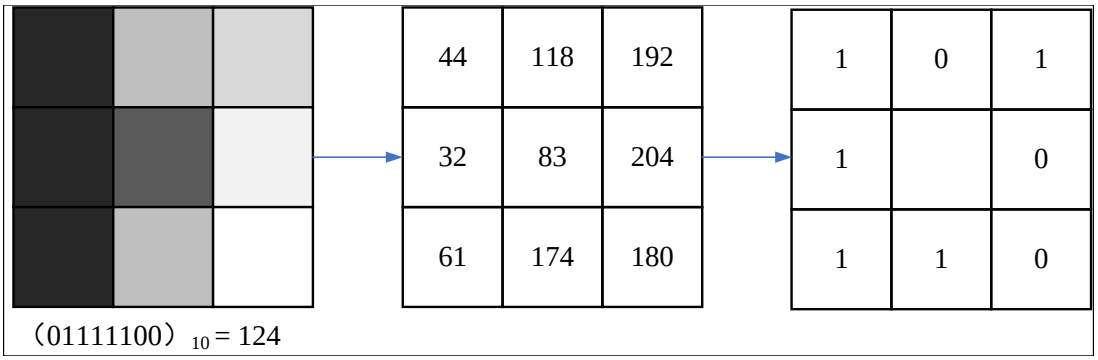

Figure 2. Extraction process of LBP

Since the original LBP operator has a small and fixed scale range and cannot be adapted to different scales as well as different frequency textures, the LBPHF operator is used in this paper. Based on the rotationally invariant LBP, the LBPHF operator makes use of the Fourier operator to characterise the rotational displacements, which is able to address the torsional offsets arising from the target motion while combining the complementary information of sign and magnitude. The advantage of the LBPHF operator over the original LBP operator is that it takes into account the frequency characteristics of the image, rather than just the spatial characteristics, and is therefore able to better capture and characterise the texture and shape characteristics of the image. LBPHF can be defined as:

$$LBP^{n2} - HF\left(n_1, n_2, u\right) = H\left(n_1, u\right)\overline{H\left(n_2, u\right)} \tag{6}$$

where $1 \leq n_1, n_2 \leq P-1$ and $1 \leq u \leq P-1, P$ are the number of feature points in the neighbourhood; $\overline{H(n_2, u)}$ is the conjugate of $H(n_2, u)$.

$$H'\left(n_1, u\right)\overline{H\left(n_2, u\right)} = H\left(n_1, u\right)e^{-i2\pi ua/P}\overline{H\left(n_2, u\right)}e^{-i2\pi ua/P} = H\left(n_1, u\right)\overline{H\left(n_2, u\right)} \tag{7}$$

where the discrete Fourier transform (DFT) is used to compute $H(n, u)$.

$$H(n,u) = \sum_{r=0}^{P-1} h_I\left(U_p(n,r)\right) e^{-i2\pi ur/P} \tag{8}$$

where $U_p(n,r)$ is a special uniform pattern, (n,r) in which n is the number of 1-bits in the uniform pattern, r is the rotation angle, the value of the histogram h_l is the number of occurrences of the uniform pattern $U_p(n,r)$ on the image I, and $H(n,\cdot)$ is the n-th row of the histogram of the Fourier transform of the histogram.

This work uses LBPHF as a general framework where $U_p(n,r)$ can be replaced by any uniform pattern feature. Using this as a basis, a new LBPHF descriptor can be obtained, which is equivalent to the complete local binary mode. The symbolic computation for the LBPHF is identical to the formulae used to compute the basic LBP. The magnitude is calculated as follows.

$$LBP_M^{P,R}(x,y) = \sum_{i=0}^{P-1} u\left(|g_{s_i} - g_c| - c\right) 2^i \tag{9}$$

where c is an adaptively determined threshold that can be set to an average value over the entire image.

Finally, the two parts were brought into Equation (9), cascaded with the resulting Fourier histograms to obtain the LBPHF combining sign and magnitude. The LBPHF extraction method does not use a block structure, but rather directly splits the image into four parts. In each part a cellular grid of 8×8 was used for segmentation. After extracting the features, the feature normalisation cascade was performed and the normalisation criteria were the same as those used for HOG feature extraction, and finally the HOG-LBPHF features were obtained.

2.2. **Fusion of features.** Currently, there are many feature fusion methods and all of them have good results for fusion, but most of them face a problem: when the detection window is larger than the size of the target in the database, and if an image contains two targets and the targets are in different regions, it may produce overlapping windows for different targets.

So in this paper, the non-maximum suppression (NMS) algorithm is used to fuse the overlap detection results. Firstly, the scores that are not positive samples are set to zero. Hard truncation is used to fuse the overlapping windows that are close to each other should be fused together, the reason is that it can reduce the time spent on mean drift later, and better results can be obtained without any adjustment.

$$t_{hc}(w) = \begin{cases} 0, & \text{if } w < c \\ w - c, & \text{if } w \geq c \end{cases} \tag{10}$$

The (x, y, scale) is mapped to a 3-D space where scale is the scale so that a number of pyramid structures are formed. Subsequently a mean shift model (mean shift model) is applied to each of the detections and the clustering results obtained are the final detections.

2.3. **Feature dimensionality reduction.** After the above HOG-LBPHF feature extraction is completed, a dimensionality reduction operation is performed using the Kernel Principal Component Analysis (KPCA) method to downscale this joint feature to a relatively low dimension to obtain the input features for target tracking. This is because the initial HOG feature is 36 dimensional and consists of four different normalisations of the 9-direction gradient histogram. However, 36-dimensional features are still too large

in dimension for target tracking and are prone to cause tracking drift which can lead to tracking failure.

KPCA is obtained by nonlinear extension of Principal Component Analysis (PCA), which can effectively deal with nonlinear features of data. The KPCA method uses a nonlinear embedding mapping, which is a nonlinear generalisation of the PCA method, and thus can effectively deal with nonlinear problems.

The traditional PCA method selects the data method with the largest variance as the projection direction, and this feature limits the application range of traditional PCA. When the original data feature space is linearly divisible, the traditional PCA method is able to downscale and classify these data, while when the original data feature space is linearly indivisible, the downscale and classification performance of the traditional PCA method will be greatly reduced. To address this shortcoming of traditional PCA, KPCA maps the features in the low-dimensional space to the high-dimensional Hilbert space by using kernel functions, and then performs linear PCA operations in the high-dimensional space to achieve data reduction and classification.

KPCA maps the samples into the high-dimensional feature space by nonlinear mapping, let the sample set be $X = \{x_1, x_2, \cdots, x_N\} \in \mathbb{R}^d$, the nonlinear map is $\emptyset$, and the high-dimensional feature space is F.

$$\emptyset : x \in X \to \emptyset(x) \in F \tag{11}$$

where $\emptyset(x)$ is the sample corresponding to x in the high-dimensional feature space F, called the kernel sample [28].

After projecting the sample set X into the feature space F, we obtain the sample set in the high-dimensional feature space $X' = \{\emptyset(x_1), \emptyset(x_2), \cdots, \emptyset(x_N)\}$, and PCA downscaling of X', i.e., to get the KPCA downscaling result of the original data X.

Assume that the sample set X' in the high-dimensional feature space F satisfies the centrality condition.

$$\sum_{i=1}^{N} \emptyset(x_i) = 0 \tag{12}$$

where m is the total number of data samples, then the covariance matrix of the sample set X' in the high-dimensional feature space F is:

$$C = \frac{1}{N} \sum_{i=1}^{N} \emptyset(x_i)\emptyset(x_i)^T \tag{13}$$

Solve for the eigenvalues and eigenvectors of C in the high-dimensional feature space F.

$$\lambda u = Cu \tag{14}$$

where λ is the eigenvalue of C and u is the corresponding eigenvector.

Since the mapping transformation $x \to \emptyset(x)$ is unknown and the high-dimensional feature space F is not visible, it is not possible to solve Equation (14) directly. According to the renewable kernel theory, $u\cdot$ can be obtained by a linear combination of feature space samples.

$$u = \sum_{i=1}^{N} a_i \emptyset(x_i) \tag{15}$$

where $a_i (i = 1, 2, \cdots, N)$ are the coefficients.

Define a kernel matrix of $N \times N$.

$$K_{ij} = K(x_i, x_j) = \emptyset(x_i)^T \emptyset(x_j) \tag{16}$$

where $K(x_i, x_j)$ is called the kernel function, which, combined with Equation (14) and Equation (15), leads to

$$\lambda' \alpha = \mathbf{K} \alpha \tag{17}$$

where $\lambda' = N\lambda$ is the eigenvalue of the kernel matrix $\mathbf{K}$ and α is the eigenvector corresponding to λ'.

After the feature extraction is completed using KPCA to perform dimensionality reduction operations on the high dimensional original features, the kernel function is selected as the radial basis function (Radial basis function), the expression of which is shown below:

$$K_{rbf} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \tag{18}$$

With KPCA, the linearly indivisible data in the original feature space can be projected into the high-dimensional feature space and then downscaled (or reconstructed) [29], so as to achieve linear divisibility in the new feature space. Figure 3 shows a sample of linearly differentiable data in the original 2D space after dimensionality reduction and reconstruction using KPCA. The left figure shows two types of concentric circle data with radii of 5 and 10. It is easy to see that these two types of data are linearly indivisible in the original feature space, and the right figure is the data reconstructed after using KPCA to downscale these two types of data. In the reconstructed feature space, these two types of data are linearly divisible, and the kernel function used in the downscaling process is the radial basis function. The parameter $2\sigma^2$ takes the value of 9.1208.

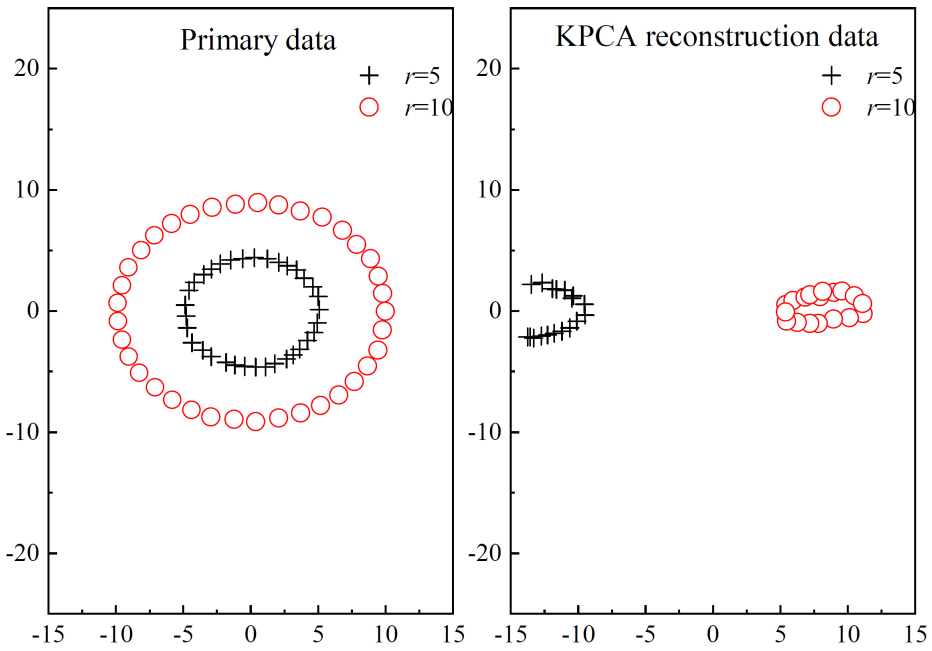

Figure 3. Instance of KPCA dimension reduction and reconstruction

3. **Improved video target tracking method based on SSVM.**

3.1. **Sample selection.** The essence of SVM is actually to find the problem of partitioning two classes a hyperplane $f(x) = w^T x + b$ such that the distance of all points from the hyperplane is kept as maximum as possible. An optimal classification plane. This is shown in Figure 4.

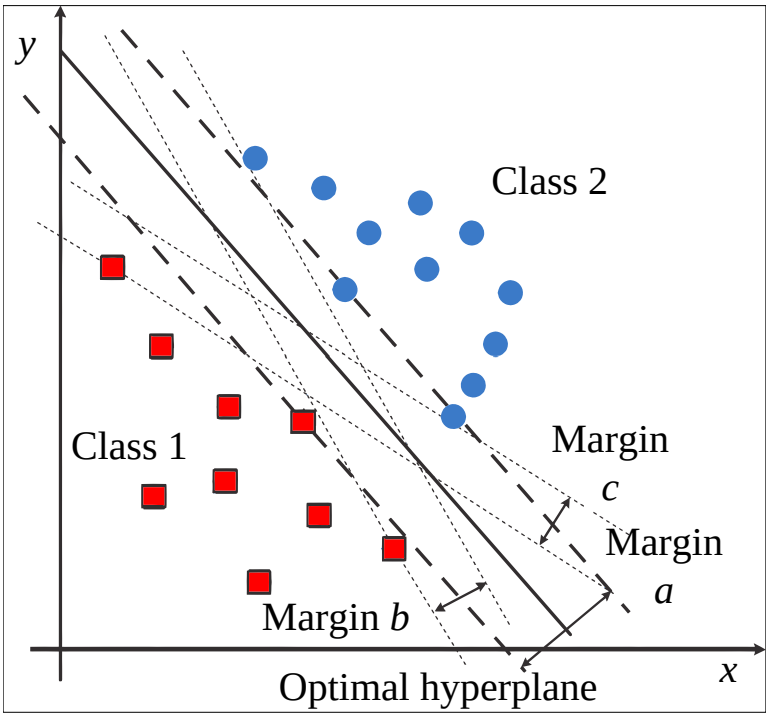

Figure 4. The optimal separating hyperplane

In this paper, we use the sample with the highest matching value as the positive sample to update the support vector, to get the positive sample, we first need to collect the sample, the collection method is the proximity region search method, we select the sample in the two-dimensional plane according to the way of $Y = \{(x, y)|x^2 + y^2 < r^2\}$, in this paper, $r = 30$, and we compute the feature vectors of the sample, and match with the support vector, this paper mainly uses Haar features, which are generated in the whole area of the target, using grey histogram and integral histogram as the basis of judgment, which can effectively reduce the feature computation time. Matched with the support vector, this paper mainly uses Haar features, generated in the target overall region, using grey scale histogram and integral histogram as the basis of judgement, can effectively reduce the feature calculation time. The expression representation of the matching function is shown as follows.

$$F(x, y) = \sum_i \beta_i \Phi(x) * \Phi(x_i) \tag{19}$$

where β_i is the weight of each support vector, x is the test feature vector, and x_i is the support vector.

3.2. **Support vector update.** The traditional method uses the highest matching value samples to update, and such samples are given in each frame, but when the target is occluded, the sample with the highest matching value is often not the target or far away from the target, and at this time, the support vectors are still updated with positive samples at a time, which will make the tracking accuracy decrease or even tracking drift.

In this paper, positive samples are selected using inter-frame matching value fluctuation as shown below:

$$|\alpha F_n - \beta F_{n-1}| < T_1,\ n \geq 2 \tag{20}$$

where α and β are weights, indicating the degree of importance of the current frame and the previous frame, here both take 1. T_1 is the value of the difference between the highest matching value of the previous frame and the current frame.

Since the match value reflects the similarity with the target, if the fluctuation of the match value suddenly becomes large, it means that the target is abnormal, and then we can not use the sample corresponding to the highest match value directly as a positive sample to update the support vector, and use the Kalman filter to correct the tracking results.

When the fluctuation value meets the threshold value T_1, the prediction target of the Kalman filter is used as the reference target, and the radius search method is used to search for samples as the new support vectors for SVM training. The search and update methods are: (1) take the current sample with the highest matching value as the centre, and stagger the points with a certain radius; (2) form a sample net group according to the method of equal division of radius and angle.

3.3. **Kalman filter correction strategy.** In the initialisation phase of target tracking, the first frame is used to calibrate the moving target by human calibration or automatic detection. A radius search is used to select the samples so as to obtain the first set of human calibrated samples $\{x_i, y_i\}_{i=1}^{N}$.

In the subsequent frames, firstly, we obtain the set of samples according to the proximity search method at the prediction location of the previous frame, calculate the matching value between the samples and the target, and update the support vectors for the samples that meet the threshold value of T_1, and at the same time, update the Kalman filter with the coordinates of the samples. If none of the extracted samples meets the threshold value T_1, then stop updating the support vectors and use the Kalman prediction correction strategy.

The Kalman filter estimates the position of the target from information about the target's dynamics, while eliminating the effects of noise. The estimated target position can be the current frame (filtering), the next frame (prediction), or the previous frame (interpolation or smoothing). In this paper, we use a combination of filtering and prediction.

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} < T_2 \tag{21}$$

where (x_i, y_i) is the SVM prediction coordinate, (x_j, y_j) is the Kalman prediction coordinate, and T_2 is the Euclidean distance determination between the two.

In this work, the mentioned threshold range matching fluctuation value T_1 is 0.25 and the Euclidean distance T_2 between SVM predicted coordinates and Kalman predicted coordinates is 8.

4. Experimental results and analyses.

4.1. **Testing of network video sequences.** In order to visually verify the tracking effect of the tracking algorithm in this paper, the experimental runtime environment is Matlab R2017a. The first test video sequences used are all from the web. The single image size of the videos is 360 × 240 and the frame rate is 30 fps. All the videos are manually labelled at the initial frame. The results of the experimental part of the target tracking are shown in Figure 5. Because the moving human body is very fast and the

athletes are similar in appearance and height, so this all brings difficulties in tracking, and all of them are jittery during the tracking period, and even the phenomenon of drift, but after a short period of time to learn all of them can be well overcome.

(a) 50 frames

(b) 100 frames

Figure 5. Partial effects of target tracking experiments

4.2. **Video target tracking performance comparison.** In order to further validate the performance of the proposed KPCA-SSVM algorithm for video target tracking, example simulations are carried out and the dataset is shown in Table 1.

From Table 2, it is known that the tracking overlap rate of the 4 types of video sets is maintained between [0.76, 0.81], which indicates that the KPCA-SSVM algorithm has a high degree of overlap between the target borders and the labelled frames predicted by the video target tracking, and it has a strong performance in adaptively tracking the target. And the tracking speed is maintained at about 30 fps for the 4-class video set.

Table 1. Video Collection

Sample name	Number of training videos	Number of test videos
GOT-10k	7823	2015
VOT2018	1632	335
UAV	5356	1013
OTB100	80	20

Table 2. Tracking performance of KPCA-SSVM

Video collection	Average overlap rate	Tracking speed (fps)
GOT-10k	0.763	29
VOT2018	0.782	37
UAV	0.779	33
OTB100	0.805	34

In order to verify the effect of KPCA on the video target tracking performance of SSVM, the tracking performance of four types of video sets are simulated using SSVM algorithm and KPCA-SSVM algorithm respectively. After KPCA processing of the video target features, the mapped features are used for SSVM to complete the target tracking by target classification, and its tracking accuracy rate is significantly improved, which is due to the fact that the KPCA of the video target features reduces the multidimensional feature correlation, and at the same time reduces the interference of video noise signals on the original video target features, which effectively improves the tracking accuracy rate of the SSVM on the video target. After the feature processing of KCPA, the video target tracking success rate of SSVM also has a large increase, which is mainly due to the fact that after the video target feature processing of KPCA, the feature dimensions are reduced and the feature redundancy is optimised, so that the tracking classification of SSVM is more effective and it is easier to track to the desired target.

Table 3. Tracking performance of SSVM and KPCA-SSVM

Video collection	Arithmetic	Average overlap rate	Rate (fps)
GOT-10k	SSVM	0.738	17
	KPCA-SSVM	0.843	27
VOT2018	SSVM	0.773	23
	KPCA-SSVM	0.862	35
UAV	SSVM	0.761	20
	KPCA-SSVM	0.859	32
OTB100	SSVM	0.783	23
	KPCA-SSVM	0.885	31

From Table 3, the KPCA-SSVM algorithm has improved the average overlap rate and tracking speed compared to SSVM. For all the given video sets, the KPCA-SSVM algorithm performs better than the SSVM algorithm. On all four video sets (GOT-10k, VOT2018, UAV, and OTB100), the average overlap rate of KPCA-SSVM is significantly higher than that of SSVM.This indicates that the KPCA-SSVM algorithm is more accurate in tracking targets. In addition to the average overlap rate, the rate (fps) of KPCA-SSVM also exceeds that of SSVM. the highest rate occurs in the VOT2018 video set at 35 fps, while SSVM has a rate of 23 fps in the same video set. this means that

KPCA-SSVM not only provides more accurate tracking, but also processes video frames faster.

In order to further compare the performance of KPCA-SSVM algorithm with other target tracking algorithms, SVM-Meanshift algorithm, FasterMDNet algorithm, RT-MDNet algorithm and KPCA-SSVM algorithm are used to simulate the performance of 4 types of video sets, the GOT-10k set for example, which is shown as Table 4.

Table 4. Tracking performance of four algorithms (GOT-10k Set)

Algorithm	**Accuracy**	**Success rate**	**Average overlap rate**	**Rate (fps)**
SVM-Meanshift	0.743	0.917	0.595	27
FasterMDNet	0.745	0.912	0.606	24
RT-MDNet	0.841	0.955	0.711	21
KPCA-SSVM	0.867	0.98	0.843	31

For the four types of video sets, the KPCA-SSVM algorithm outperforms the other three algorithms in all four metrics, while the other three algorithms have their own strengths and weaknesses in a single metric. In terms of video tracking rate, the SVM-Meanshift algorithm is second only to KPCA-SSVM, which is mainly due to the fact that the classification efficiency of SVM is generally higher than the network optimisation of MDNet. The comparison reveals that the KPCA-SSVM algorithm better solves the dual optimisation problem of video target tracking accuracy and rate.

5. **Conclusion.** In this work, a video target tracking method based on KPCA-SSVM is proposed. LBPHF is used as a general framework and a new LBPHF descriptor is obtained based on it. The two features are fused using the non-maximum suppression (NMS) method to obtain the joint HOG-LBPHF feature. After extracting the HOG-LBPHF joint features of the video target, it is mapped to a feature vector more favourable for target classification by KPCA. The inter-frame target matching fluctuation value is used to judge whether the target appears abnormal or not, and combined with Kalman filter to predict the target location in the next frame. The experimental results show that the KPCA-SSVM algorithm achieves high tracking performance in tracking studies of all four types of public video sets. The subsequent study will consider the use of intelligent algorithms to optimally solve the core parameters of SSVM from the perspective of optimised solution of SSVM, in order to further improve the applicability of the KPCA-SSVM algorithm in video target tracking applications.

REFERENCES

[1] H. Han, Y.-S. Ding, K.-R. Hao, and X. Liang, "An evolutionary particle filter with the immune genetic algorithm for intelligent video target tracking," *Computers & Mathematics with Applications*, vol. 62, no. 7, pp. 2685-2695, 2011.

[2] G. S. Walia, and R. Kapoor, "Intelligent video target tracking using an evolutionary particle filter based upon improved cuckoo search," *Expert Systems with Applications*, vol. 41, no. 14, pp. 6315-6326, 2014.

[3] V. Cevher, A. C. Sankaranarayanan, J. H. McClellan, and R. Chellappa, "Target tracking using a joint acoustic video system," *IEEE Transactions on Multimedia*, vol. 9, no. 4, pp. 715-727, 2007.

[4] T. Nawaz, F. Poiesi, and A. Cavallaro, "Measures of effective video tracking," *IEEE Transactions on Image Processing*, vol. 23, no. 1, pp. 376-388, 2013.

[5] M. Wan, G. Gu, W. Qian, K. Ren, X. Maldague, and Q. Chen, "Unmanned aerial vehicle video-based target tracking algorithm using sparse representation," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9689-9706, 2019.

[6] M. Nallasivam, and V. Senniappan, "Moving human target detection and tracking in video frames," *Studies in Informatics and Control*, vol. 30, no. 1, pp. 119-129, 2021.

[7] H. Shao, J. Shen, Z. Zhang, and H. Liu, "Research and analysis of video image target tracking algorithm based on significance," *International Journal of High Performance Systems Architecture*, vol. 8, no. 1-2, pp. 82-93, 2018.
[8] Y. Wang, T. Wang, G. Zhang, Q. Cheng, and J.-q. Wu, "Small target tracking in satellite videos using background compensation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 10, pp. 7010-7021, 2020.
[9] L. Wang, T. Liu, G. Wang, K. L. Chan, and Q. Yang, "Video tracking using learned hierarchical features," *IEEE Transactions on Image Processing*, vol. 24, no. 4, pp. 1424-1435, 2015.
[10] F. Zhang, T.-Y. Wu, J.-S. Pan, G. Ding, and Z. Li, "Human motion recognition based on SVM in VR art media interaction environment," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, 40, 2019.
[11] F. Zhang, T.-Y. Wu, and G. Zheng, "Video salient region detection model based on wavelet transform and feature comparison," *EURASIP Journal on Image and Video Processing*, vol. 2019, 58, 2019.
[12] K. Wang, C.-M. Chen, M. S. Hossain, G. Muhammad, S. Kumar, and S. Kumari, "Transfer reinforcement learning-based road object detection in next generation IoT domain," *Computer Networks*, vol. 193, 108078, 2021.
[13] D. Wang, L. Shi, and P. A. Heng, "Automatic detection of breast cancers in mammograms using structured support vector machines," *Neurocomputing*, vol. 72, no. 13-15, pp. 3296-3302, 2009.
[14] G. Blanchard, O. Bousquet, and L. Zwald, "Statistical properties of kernel principal component analysis," *Machine Learning*, vol. 66, pp. 259-294, 2007.
[15] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1631-1643, 2005.
[16] F. Bousetouane, L. Dib, and H. Snoussi, "Improved mean shift integrating texture and color features for robust real time object tracking," *The Visual Computer*, vol. 29, pp. 155-170, 2013.
[17] D. Weiskopf, T. Schafhitzel, and T. Ertl, "Texture-based visualization of unsteady 3d flow by real-time advection and volumetric illumination," *IEEE Transactions on Visualization & Computer Graphics*, vol. 13, no. 03, pp. 569-582, 2007.
[18] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 10, pp. 2002-2015, 2014.
[19] G. Li, Q. Huang, S. Jiang, Y. Xu, and W. Zhang, "Online learning affinity measure with CovBoost for multi-target tracking," *Neurocomputing*, vol. 168, pp. 327-335, 2015.
[20] S. Zhang, X. Yu, Y. Sui, S. Zhao, and L. Zhang, "Object tracking with multi-view support vector machines," *IEEE Transactions on Multimedia*, vol. 17, no. 3, pp. 265-278, 2015.
[21] D. Zhang, L. Jiao, X. Bai, S. Wang, and B. Hou, "A robust semi-supervised SVM via ensemble learning," *Applied Soft Computing*, vol. 65, pp. 632-643, 2018.
[22] J. Li, X. Zhou, S. Chan, and S. Chen, "Object tracking using a convolutional network and a structured output SVM," *Computational Visual Media*, vol. 3, pp. 325-335, 2017.
[23] W. J. Lee, G. P. Mendis, M. J. Triebe, and J. W. Sutherland, "Monitoring of a machining process using kernel principal component analysis and kernel density estimation," *Journal of Intelligent Manufacturing*, vol. 31, pp. 1175-1189, 2020.
[24] M. Sultana, T. Ahmed, P. Chakraborty, M. Khatun, M. R. Hasan, and M. S. Uddin, "Object detection using template and HOG feature matching," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 7, pp. 233-238, 2020.
[25] A. Baâzaoui, M. Abderrahim, and W. Barhoumi, "Dynamic distance learning for joint assessment of visual and semantic similarities within the framework of medical image retrieval," *Computers in Biology and Medicine*, vol. 122, 103833, 2020.
[26] C. Guo, M. Cai, N. Ying, H. Chen, J. Zhang, and D. Zhou, "ANMS: attention-based non-maximum suppression," *Multimedia Tools and Applications*, vol. 81, no. 8, pp. 11205-11219, 2022.
[27] S. Karanwal, and M. Diwakar, "OD-LBP: Orthogonal difference-local binary pattern for Face Recognition," *Digital Signal Processing*, vol. 110, pp. 102948, 2021.
[28] B. M. S. Hasan, and A. M. Abdulazeez, "A review of principal component analysis algorithm for dimensionality reduction," *Journal of Soft Computing and Data Mining*, vol. 2, no. 1, pp. 20-30, 2021.
[29] E. O. Omuya, G. O. Okeyo, and M. W. Kimwele, "Feature selection for classification using principal component analysis and information gain," *Expert Systems with Applications*, vol. 174, pp. 114765, 2021.