

## Integral region-based covariance tracking with occlusion detection

Ruhan He · Bing Yang · Nong Sang · Yongsheng Yu · Geli Bai · Jizi Li

Published online: 22 January 2014  
© Springer Science+Business Media New York 2014

**Abstract** Covariance tracking has achieved impressive successes in recent years due to its competent region covariance-based feature descriptor. Although adopt fast integral image computation, covariance tracking's brute-force search strategy is still inefficient and it possibly leads to inconsecutive tracking trajectory and distraction. In this work, a generalized, adaptive covariance tracking approach with novel integral region computation and occlusion detection is proposed. The integral region is much faster than integral image and adaptive to the tracking target and tracking condition. The adaptive search window can be adjusted dynamically by simple occlusion detection. The integral image and the global covariance tracking can be seen just a special case of integral region and the proposed approach, respectively. The proposed approach unifies the local and global search strategies in an elegant

---

R. He  
College of Mathematics and Computer Science, Wuhan Textile University, Wuhan 430073, China  
e-mail: heruhan@hotmail.com

B. Yang (✉)  
School of Education, HuBei University, Wuhan 430062, China  
e-mail: hubuyangbing@gmail.com

N. Sang  
Institute for Pattern Recognition and Artificial Intelligence,  
Huazhong University of Science and Technology, Wuhan 430074, China  
e-mail: nsang@hust.edu.cn

Y. Yu  
The Green Building Materials and Manufacturing Engineering Research Center of Ministry of Education, Wuhan University of Technology, Wuhan 430070, China  
e-mail: yongshengyu@whut.edu.cn

G. Bai  
College of computer and information engineering, Inner Mongolia Agricultural University,  
Huhhot 010018 China

J. Li  
School of Management, Wuhan Textile University, Wuhan 430073, China

way and smoothly switches between them according to the tracking conditions (i.e. occlusion distraction or sudden motion) which are judged by occlusion detector. It gets much better efficiency, robustness of distraction, and stable trajectory by local search in normal steady state, while obtains more abilities for occlusion and re-identification by enlarged search window (until to global search) in abnormal situation at the same time. Our approach shows excellent target representation ability, faster speed, and more robustness, which has been verified on some video sequences.

**Keywords** Integral region · Covariance tracking · Symmetric positive definite matrix · Occlusion detection

## 1 Introduction

Object tracking is a challenging problem in computer vision, which tries to determine the position of an object in the image plane as it moves around a scene. One challenge lies in handling the appearance variability of a target, such as pose change, shape deformation, illumination change, camera viewpoint change, occlusion, background clutter, distraction, etc. Consequently, it is imperative for a robust tracking algorithm to model such appearance variations while ensuring real-time and accurate performance at the same time.

There have been many approaches for object tracking in recent years [2, 6, 12, 14–17, 21, 22, 27, 33, 35, 42–44, 46–49]. A detailed summary before 2006 can be found in [45] and an object tracking benchmark including many latest tracking algorithm is proposed recently [40]. Consequently, one representative approach is covariance tracking [31], which represents targets by covariance matrices. The covariance matrix offers competent, compact, informative, and empirically beneficial feature descriptors, which fuses multiple modalities and features in a natural and elegant way and captures both statistical and spatial properties of the object appearance. Due to its global nature, the algorithm can recover from full occlusions or large movements. Although an integral image-based method [34] is used to improve the speed, it is still time consuming due to the global brute-force searching strategy. Meanwhile, it is possibly leads to inconsecutive tracking trajectory and susceptible to background clutter, false positives, and outliers distraction.

This strategy of global search to find the best matching image patch (to the model covariance matrix) is slow for many real-time applications and does not scale well to larger images and higher dimensional lattices (e.g., 3D tracking). Moreover, searching the entire image is prone to errors that can be caused by distractions from similar objects present in the scene.

On the other hand, based on the continuity assumption of the moving object in video, some local search-based tracking algorithms also attracted a lot of attention, such as Mean Shift (MS) [10, 11]. MS tracking is very fast due to its local searching strategy. However, it suffers from a number of weaknesses: it is prone to failure when there exist large motions or occlusions for the target since only local searching is carried out; it is prone to be distracted by other objects similar to the target and it lacks a mechanism for encoding the spatial layout of the object's colors. The tracking approaches based on the different two types of search strategies obviously have their merits and drawbacks respectively. Based on the simple idea that: a simple and efficient local search strategy should be used in normal steady situation while a global search strategy should be utilized for abnormal, difficult situation such as long-term occlusion, sudden or fast motion. We try to find a way to naturally fuse and uniform these two.

Therefore, motivated by the previous works in [31, 34], a generalized, adaptive covariance tracking approach with integral region computation and occlusion detection is proposed in this paper. The size of integral region is adaptive to the tracking target's size and tracking condition. The integral image can be seen a special case of integral region. The search window in the proposed approach is adaptive and flexible, which is adjusted by occlusion detection. Our approach takes advantage of the two types of search strategies, i.e. global and local, for tracking. Local covariance tracking is used in normal steady state, which results to fast computation and robustness for distraction and stable trajectory, while the search window can be dynamically enlarged (until to global search) by occlusion degree and time in occlusion condition, which gives more abilities for occlusion and re-identification. The two strategies are unified in the proposed approach in an elegant way and smoothly switched by the occlusion detector. Moreover, the new integral region method is much faster for the computation of regional covariance and adapts to the local search mode which is different from the integral image method used in [31]. Our approach obtains excellent target representation ability, faster speed, and more robustness.

The proposed approach simultaneously utilizes the discriminative ability of covariance descriptor and the fast computation of local search mode, robustness of the adaptive covariance tracking. The contributions of this paper are: 1) propose a novel fast region covariance computation based on integral region, different from the integral image; 2) generalize covariance tracking to adapt to different size of regions, which can regress into the global covariance tracking; 3) smoothly switch between the global and the local methods with occlusion detector.

The remainder of the paper is organized as follows. The related work is presented in Section 2. We describe our proposed approach in Section 3. We propose an experimental evaluation for different tracking algorithms. Finally, we summarize and give concluding remarks.

## 2 Related work

There have been lots of state-of-art visual tracking methods in recent years, such as Viola-Jones [36], Struct [14], GMCP [46], OpenTLD [18], VTD[20], LI[24], Online Boost Learning [13], P-N Learning [19], Multiple Instance Learning [5], Incremental Visual Learning [32], Part-based Tracking [1, 33], etc. A detailed survey on object tracking before 2006 can be found in [45] and an object tracking benchmark including many latest tracking algorithm is proposed in [40]. Tracking algorithms can be broadly classified as either filtering and data association or target representation and localization methods [35]. Kalman and particle filters belong to the former class.

Mean shift tracking [10, 11] is a class of localization-based tracking algorithm, which has got popularity due to its computational efficiency and ease of implementation. The mean-shift algorithm is a robust non-parametric probability density gradient estimation method [9]. Mean-shift tracking [10, 11] tries to find the region of a video frame that is locally most similar to a previously initialized object model. The target model and the target candidate region to be tracked are represented by appearance histograms. A gradient ascent procedure is used to move the tracker to the location that maximizes a similarity score between the model and the current image region. Similarity between the histograms is evaluated as the Bhattacharyya coefficient between the two distributions. However, the basic mean shift tracking algorithm assumes that the target representation is sufficiently discriminative

against the background. This assumption is not always true especially when the tracking is carried out in a dynamic background such as surveillance with a moving camera.

Moreover, one limitation of these localization-based methods is that the inter-frame displacement of the object must be small; otherwise, the tracker will lose its target. Collins [8] proposed an explicit scale-space approach for tracking objects to resolve the unreliable scale change issue. Nguyen et al. [25] recovered the scale of the object by an expectation maximization algorithm. Porikli and Tuzel [30] initialized multiple trackers around the expected location of the object to overcome the difficulty. Some others used multiple-part models and background exclusion histograms [23, 26, 41] as the target and candidate models to alleviate the lack of spatial structure, and improve the discriminative ability of the object representations.

More recently, the covariance region descriptor proposed in [34] has been proven to be robust and versatile. The use of covariance features as target representation for tracking was firstly proposed by [31]. Consequently, it is widely used in tracking [7, 37–39]. The covariance matrix of features extracted from an image patch enables a compact representation of both the spatial and the statistical properties of the object, and its dimensionality is small. The tracker performs an exhaustive search in the image by comparing the given covariance model to the covariance matrix at each location using an appropriately defined distance metric. The location whose covariance matrix is most similar to the target model is assigned to be the new target position in the image. The computational cost is high because the target localization process is formulated as an expensive exhaustive searching. Moreover, the similarity measure in [31] is adopted from [28], which is an affine invariant metric. Using the covariance descriptor is time consuming, computationally inefficient, easily affected by background clutter, and ineffective over occlusions.

The objective of our approach in this paper is to take advantage of the two types of search strategies, i.e. global and local, for tracking to get a better accurate, robust and efficient tracking.

### 3 The proposed approach

In the following, we first introduce the region covariance and how to compute it quickly by integral region; then describe the manifold distance for the covariance matrices, which is different from the usual Euclidean distance; therefore, give the occlusion detector and how to switch between local and global search; finally, describe the adaptive covariance tracking approach.

#### 3.1 Fast region covariance computation based on integral region

Using covariance matrix (CM) as a region descriptor is an elegant and simple solution to integrate multiple image features. It has many advantages, namely: 1) CM indicates both spatial and statistical properties of the objects; 2) it provides an elegant means to combine multiple modalities and features; 3) it is capable of relating regions of different sizes.

For fast calculation of region covariance, reference [34] proposed an integral image method. Here, based on the local assumption, we follow a similar idea to define an integral region for the calculation of region covariance, which is much faster generally. Moreover, it can be regressed into integral image at certain condition, i.e. integral image is a special case of integral region.

Let  $I$  be a one dimensional intensity or three dimensional color image, and  $F$  be the  $H \times W \times d$  dimensional feature image extracted from  $I$ ,  $F(x, y) = \phi(I, x, y)$ , where  $\phi$  can be any mapping such as color, texture, shape, gradients, filter responses, etc. Let  $\{f_i\}_{i=1}^N$  be the  $d$ -dimensional feature points inside a given rectangular region  $R$  of  $F$ . Region  $R$  is represented by the  $d \times d$  covariance matrix of the feature points, i. e.

$$C_R = \frac{1}{N - 1} \sum_{i=1}^N (f_i - u)(f_i - u)^T \tag{1}$$

where  $N$  is the number of pixels in region  $R$  and  $u$  is the mean of the feature points. Elements  $(i, j)$  of  $C_R$  represents the correlation between features  $i$  and  $j$ . When the extracted  $d$ -dimensional feature includes the pixels coordinate, the covariance descriptor encodes the spatial information of features. With the help of integral region, the covariance descriptor can be calculated efficiently.

Let  $P$  be the  $(x_1 - x_0 + 1) \times (y_1 - y_0 + 1) \times d$  tensor of the integral regions and  $Q$  be the  $(x_1 - x_0 + 1) \times (y_1 - y_0 + 1) \times d \times d$  tensor of the second-order integral regions.

$$P(x', y', i) = \sum_{x_0 \leq x \leq x_0 + x' + 1, y_0 \leq y \leq y_0 + y' + 1} F(x, y, i) \tag{2}$$

$$Q(x', y', i, j) = \sum_{x_0 \leq x \leq x_0 + x' + 1, y_0 \leq y \leq y_0 + y' + 1} F(x, y, i)F(x, y, j) \tag{3}$$

where  $i = 1 \dots d, 1 \leq x' \leq (x_1 - x_0 + 1), 1 \leq y' \leq (y_1 - y_0 + 1)$ . Here we take  $P(x, y)$  as  $d$ -dimensional vector and  $Q(x, y)$  as  $d \times d$  dimensional matrix.

$$P(x, y) = [P(x, y, 1) \dots P(x, y, d)]^T \tag{4}$$

$$Q(x, y) = \begin{pmatrix} Q(x, y, 1, 1) & \dots & Q(x, y, 1, d) \\ \dots & \dots & \dots \\ Q(x, y, d, 1) & \dots & Q(x, y, d, d) \end{pmatrix} \tag{5}$$

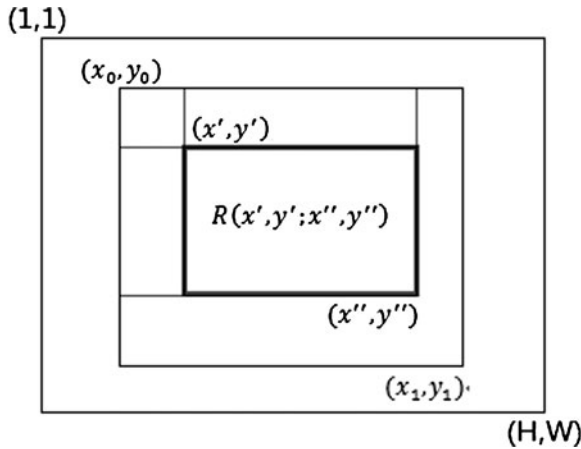
The  $Q(x, y)$  is a symmetric matrix.

Let  $R(x', y'; x'', y'')$  be the rectangular region, where  $(x', y')$  is the upper left coordinate and  $(x'', y'')$  is the lower right coordinate, as shown in Fig. 1. The covariance of the region the region  $R(x', y'; x'', y'')$  can be computed as

$$\begin{aligned} C_R(x', y'; x'', y'') &= \frac{1}{n} (Q(x'' - x_0 + 1, y'' - y_0 + 1) + Q(x' - x_0 + 1, y' - y_0) \\ &\quad - Q(x'' - x_0 + 1, y' - y_0) - Q(x' - x_0, y'' - y_0 + 1) \\ &\quad - \frac{1}{n - 1} ((P(x'' - x_0 + 1, y'' - y_0 + 1) + P(x' - x_0, y' - y_0) \\ &\quad - P(x'' - x_0 + 1, y' - y_0) - P(x' - x_0, y'' - y_0 + 1)) \\ &\quad \cdot (P(x'' - x_0 + 1, y'' - y_0 + 1) + P(x' - x_0, y' - y_0) \\ &\quad - P(x'' - x_0 + 1, y' - y_0) - P(x' - x_0, y'' - y_0 + 1))^T) \end{aligned} \tag{6}$$

where  $n = (x_1 - x_0 + 1) \times (y_1 - y_0 + 1), 1 \leq x', x'' \leq (x_1 - x_0 + 1), 1 \leq y', y'' \leq (y_1 - y_0 + 1)$ . Therefore, after constructing integral region, the covariance of any rectangular region in the scope of  $(x_0, y_0)$  and  $(x_1, y_1)$  can be computed very fast.

In most of cases, the tracking object is far smaller than the whole image, and its corresponding integral region is much smaller than the integral image of the image, thus the computation of constructing the integral region is much lower than that of constructing the



**Fig. 1** Integral region. The integral region is the rectangle defined by  $(x_0, y_0)$  and  $(x_1, y_1)$ . The covariance of rectangle  $R(x', y'; x'', y'')$  is computed within the integral region

integral image. Moreover, when  $(x_0, y_0)$  is equal to  $(1, 1)$  and  $(x_1, y_1)$  is equal to  $(H, W)$ , the integral region regresses into integral image. In short, the integral region can be seen a generalized integral image. It adapts to local search mode with dynamic regions, and dramatically reduces the time cost in constructing the integral.

### 3.2 Manifold distance for covariance matrices

Covariance matrices are typical symmetric positive definite (SPD) matrices which do not conform to Euclidean geometry, but rather forms a Riemannian manifold. Therefore, it makes defining similarity measures between covariance matrices non-trivial.

Since the Affine Invariant Riemannian Metric (AIRM)[28] used in [31] is computationally expensive, we apply the efficient Log-Euclidean Riemannian Metric (LERM) [3] in the manifold.

Let  $U \cdot S \cdot U^T$  be the singular value decomposition of a  $n \times n$  SPD (i.e.  $Sym_+^n$ ). If  $X \in Sym_+^n$ , then the matrix logarithm is

$$\log(X) = U \cdot \log(S) \cdot U^T \tag{7}$$

where the *log* acts on element-by-element along the diagonal of  $S$ . Since the singular values of a SPD matrix are always positive, the log can act along the diagonal. Similarly, let  $X$  be a symmetric matrix with SVD given  $X = U \cdot D \cdot U^T$ . The matrix exponential becomes

$$\exp(X) = U \cdot \exp(D) \cdot U^T \tag{8}$$

where the *exp* acts element-by-element along the diagonal of  $D$ . It is easy to see that this form of the exponential is defined for any symmetric matrix and the result is a SPD matrix.

Therefore, according to [4], the manifold logarithm operator for two  $n \times n$  SPD matrices  $X$  and  $Y$  is

$$\log_X(Y) = \log(Y) - \log(X) \tag{9}$$

Given a tangent  $\Delta$ , the manifold exponential operator is

$$\exp_X(\Delta) = \exp(\log(X) + \Delta) \tag{10}$$

Finally, the manifold distance between SPD matrices  $X$  and  $Y$  is given by

$$d^2(X, Y) = \| \log(Y) - \log(X) \|_2 \tag{11}$$

We build the mapping between the manifold space  $Sym_+^n$  and tangent vector space by the manifold  $\log$  operator and the  $\exp$  operator.

### 3.3 Occlusion detection and switch criteria

For occlusion detection, we investigate the return distances in the tracking. The distance becomes much larger when the tracking region cannot be matched well with the target model. We threshold the distance based on the statistical computation of the distances in previous tracking frames without occlusion. We assume it is a normal distribution  $(\mu, \sigma^2)$ , then the threshold is  $T = \mu + 3\sigma$  (according to the  $3\sigma$  principle of normal distribution). If the distance is larger than  $T$ , we conclude that there is possibly an occlusion for current frame. Therefore, we further compare the target in current frame  $f_t$  and the target in latest  $L$ -th non-occluded frame  $f_{t-L}$  to decide whether the target in  $f_t$  is occluded or not. This is achieved by checking the difference between pixel values of  $f_t$  and  $f_{t-L}$  at the same pixel is greater than a threshold (e.g. 30) or not. This procedure is described in the following equation.

$$O_{t,i} = \begin{cases} 1 & \text{if } |f_{t,i} - f_{t-L,i}| < \varepsilon \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

where  $f_{t,i}$  is the gray value at location  $i$  in frame  $f_t$ , and  $f_{t-L,i}$  is similar, and  $i$  is the location within the target.  $O_{t,i}$  is a binary map with target size. We perform a morphological closing with the structuring element in (13) on the binary map,

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{13}$$

and find the largest connected area in the map. Finally, we decide the target is occluded if the maximum area is greater than 1/4 of the target area. The occlusion detection is simple and false positive is possibly happens, but it does not matter because we just enlarge search window in the following search in case of bad situation. In the occlusion period, we predict the new object location based on the object dynamics (velocity).we keep the object window moving at the same speed and direction as before, and enlarge the search window for tracking. We use a simple adaptive velocity model. It is an average velocity of the last  $k$  instants,

$$v_t = \frac{1}{k} \sum_{n=t-k}^t |x_n - x_{n-1}| \tag{14}$$

where  $k$  is the number of the latest non-occlusion frames,  $x_n$  is the pixel location of the target center in  $n$ -th frame.

In addition, when an occlusion is detected, it will not go away for a certain period of time. For example, when the target is occluded by an object, and the object is moving away from the target, the occlusion is becoming smaller before it goes away. In our occlusion detection method, we jump over the next 2 frames after an occlusion is detected, which is used as a compensation for the added computation due to the larger search window.

When the return distance is normal again, we assume the occlusion is over, and the size of search window is also restored.

In addition, it is dangerous to update the target model when tracking is in an unsteady state. Therefore different from [31], the target model does not be updated when current frame is labeled to be occluded in case of contaminating the target model.

### 3.4 Generalized adaptive covariance tracking

Generalized adaptive covariance tracking is a flexible tracking approach. It assumes the object will appear again in the neighbor of the location of disappearing. When the neighbor is same as the whole image, it regresses into global covariance tracking. Therefore, the proposed approach can be seen a generalized covariance tracking. The size of search window can be tuned differently according to the tasks need.

We generally enlarge the search window to deal with these problems including occlusions, sudden motions and distractions, when they are detected during the tracking. Meanwhile, we simply set the search window to be same as the target region in normal steady state.

Assume the target center in current frame to be initialized at  $(\bar{x}, \bar{y})$ , and the size to be  $(W_1, H_1)$ , where  $W_1$  is width of the target and  $H_1$  is the height. The search range, i.e. search window, which is the range of the all candidate target center, is represented by  $(W_2, H_2)$ . Therefore, we need to define region  $R(x_0, y_0; x_1, y_1)$  to construct integral region, i.e.

$$(x_0, y_0) = (\bar{x}, \bar{y}) - \frac{(W_2, H_2)}{2} - \frac{(W_1, H_1)}{2} \tag{15}$$

$$(x_1, y_1) = (\bar{x}, \bar{y}) + \frac{(W_2, H_2)}{2} + \frac{(W_1, H_1)}{2} \tag{16}$$

We define a scale factor  $(s_W, s_H)$  for the search widow, i.e.

$$(W_2, H_2) = (s_W * W_1, s_H * H_1) \tag{17}$$

When our approach runs at steady state for tracking, the search range, i.e. search window, is same as the target region, i.e.  $(s_w, s_H) = (1, 1)$ . However, when we need to cope with abnormal condition, we can adjust the search window in our approach. When we enlarge the search window, it is helpful to re-identify the object after the occlusion. Therefore, when occlusion is detected we set

$$s_W = 1 + s \cdot \frac{\max(2 \cdot \text{avg}(\Delta x)_L, \max(\Delta x)_L)}{W_1} \tag{18}$$

$$s_H = 1 + s \cdot \frac{\max(2 \cdot \text{avg}(\Delta y)_L, \max(\Delta y)_L)}{H_1} \tag{19}$$

where  $\text{avg}(\Delta x)_L$  and  $\text{avg}(\Delta y)_L$  are the average  $x$  and  $y$  offsets of the target center in the latest  $L$  non-occluded frames,  $\max(\Delta x)_L$  and  $\max(\Delta y)_L$  are the maximum  $x$  and  $y$  offsets of the target center in the latest  $L$  non-occluded frames,  $s$  is an increasing factor for  $s_H$  and  $s_W$ . If  $s$  is larger, the search window enlarges faster when there is an occlusion. Otherwise, the search window enlarges slower. Base on the scale factor, we make sure that: if the target re-appears at the last target window it disappeared, we can found it, and if the target moves forwards at its fastest speed, we also can find it. This is because the new search window will include these possible positions. In addition, obviously, we need to keep  $R(x_0, y_0; x_1, y_1)$  to be in image range, and search window to be in  $R(x_0, y_0; x_1, y_1)$ .



The adaptive covariance tracking approach is shown in Algorithm 1.

---

**Algorithm 1** Adaptive Covariance Tracking Approach.
 

---

**Input:**

The covariance descriptor of initial target model,  $X_{model}$ ;  
 The estimated location of the target in the previous frame,  $x_0 = (\bar{x}, \bar{y})$ ;  
 The target size,  $(W_1, H_1)$ ;  
 The scale factor,  $(s_W, s_H)$ .

**Output:**

The new location in current frame,  $x'$ .

- 1: Get search window according to (17).
- 2: Ascertain the region to compute integral region according to (15) and (16).
- 3: Construct integral region according to (4) and (5).
- 4: According to (6), for each point  $x_i$  in search window, compute the region covariance of the corresponding region (i.e. the center point is  $x_i$ , the top-left point is  $(x', y')$  and the bottom-right point is  $(x'', y'')$ ).
- 5: The SPD manifold point  $X_i = F(x_i) = C_R(x', y'; x'', y'')$ , and calculate  $d^2(X_{model}, X_i)$  according to (11);
- 6: Get the new location  $x'$  based on (18)

$$x' = \arg \min_{x_i} (d^2(X_{model}, X_i)) \quad (20)$$

- 7: **return**  $x'$ .
- 

In other words, the proposed integral region and adaptive covariance tracking approach are dynamic based on the tracking condition. In normal steady state,  $s_W = 1$  and  $s_H = 1$ , and the size of the search window is equal to that of the target window in the previous frame. Therefore, our approach regresses into local search-based covariance tracking. However, when an occlusion is detected,  $s_W$  and  $s_H$  are bigger, and the search window and the integral region are accordingly enlarged. If the occlusion is continuously detected,  $s_W$  and  $s_H$  are bigger and bigger, until the search window is equal to the whole image, then the integral region is equal to integral image. Therefore, our approach is equal to the traditional global search-based covariance tracking in this case.

## 4 Experiments

We evaluate the proposed adaptive covariance tracking approach (AdaCov) in comparison to the standard histogram-based Mean Shift tracking algorithm (HistMS) [10, 11], the original covariance tracking algorithm (GlobalCov) [31], the local covariance tracking (LocalCov) with fixed search window, the color-based particle filter tracking algorithm (ColorPF) [29], and the gradient descent covariance tracking algorithm (GradientCov) [35]. The algorithms are compared for three aspects including accuracy, robustness, and computational efficiency.

In our experiments, the target is manually initialized. The tracking parameters are tuned on one sequence and applied to all the other sequences. Except for HistMS and

ColorPF, during the visual tracking, a  $7 - D$  feature vector is extracted for each pixel, i.e.  $(x, y, R, G, B, I_x, I_y)$ , where  $(x, y)$  is the pixel location;  $R, G, B$  are the RGB color values; and  $I_x$  and  $I_y$  are the intensity derivatives. Consequently, the covariance descriptor of a color image region is a  $7 \times 7$  symmetric matrix.

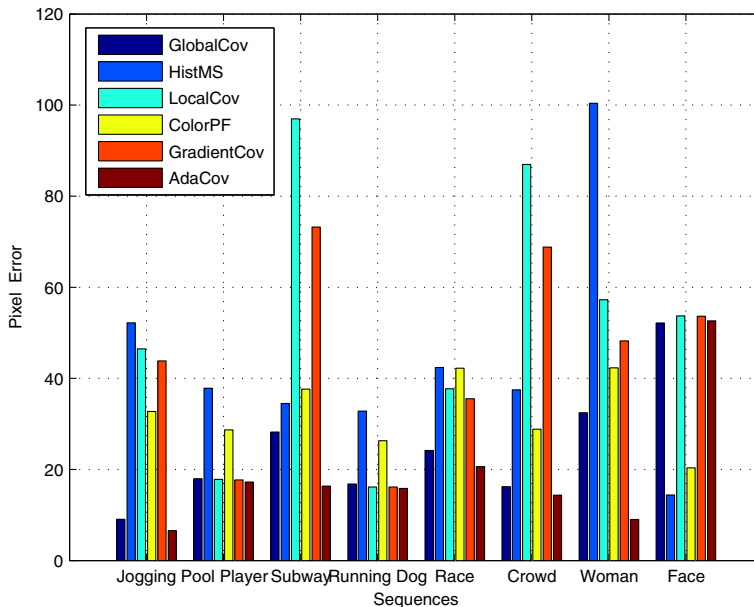
For ColorPF, we adopt the source code from reference [29], which describes the object by HSV color. For HistMS, we used a joint histogram of color-gradient features extracted from the image. At each pixel, a five dimensional feature  $(R, G, B, I_x, I_y)$  vector is extracted to form the  $Bin^5$ -dimensional quantized histogram, here we set  $Bin = 16$ .

The dataset is from [1, 31], including “Jogging”, “Pool Player”, “Subway”, “Running Dog”, “Race”, “Crowd”, “Woman” and “Face” sequences. These sequences have been widely used before in visual tracking experiments and have ground truth data for convenient comparison. There are full occlusions in “Jogging” sequence and size changes in “Race” sequence. In “Pool Player” and “Running Dog” sequences, the camera and objects are moving and the appearances are changing. In “Subway” and “Crowd” sequences, the objects have indistinctive color and insignificant texture information. There are partial occlusions in “Face” sequence, and there are partial occlusions, illumination changes and distractions in “Woman” sequence.

#### 4.1 Accuracy

First we compared the tracking accuracy of the six approaches. Figure 2 shows the tracking pixel error produced by each approach on the different sequences with respect to the ground truth.

It can be seen from Fig. 2 that HistMS, GradientCov and LocalCov are good to some sequences while bad to some other sequences. The reason lies in they are local search-based methods and they generally can not catch the target again in case of losing the target, which

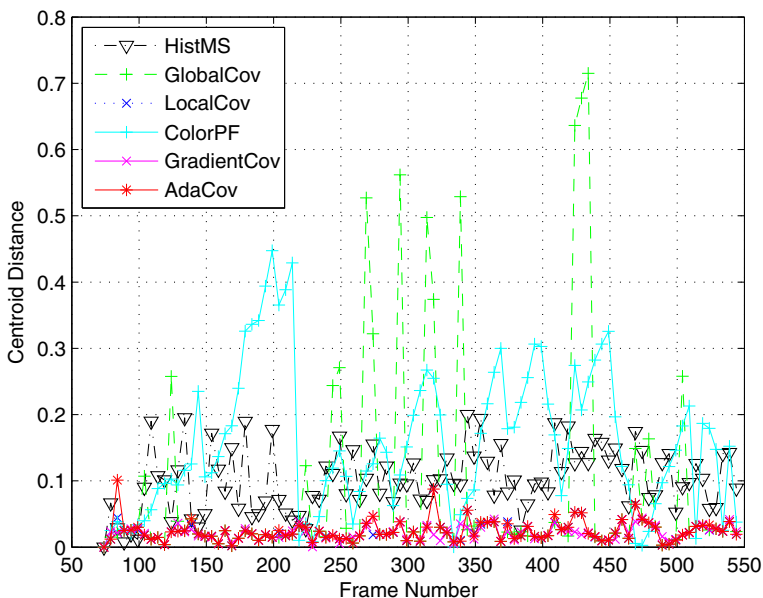


**Fig. 2** Pixel error for different sequences

often leads to very high pixel error. They can possibly get good results only if the target does not get out of the local search window. LocalCov and GradientCov result in some high tracking pixel errors (e.g. “Subway” and “Crowd” sequences) and some low tracking pixel errors (e.g. “Pool Player”, “Woman” and “Running Dog” sequences). The reason lies in their exploiting the spatiotemporal locality constraint. HistMS is not good in pixel error in most of cases due to its local search strategy and relative weak feature descriptor. However, there is an exception for HistMs on “Face” sequence where it gets the best result among all approaches. On the other hand, GlobalCov and AdaCov have the ability to overcome the full occlusion, and catch the target again even if they lose the target temporarily. GlobalCov searches the matched target in whole image, and hence it can recover from lost track and also maybe gets distracted due to similar objects, thus resulting in discontinuous tracking trajectory. ColorPF is normal in pixel error in most of cases. It can be easily distracted due to its weak feature descriptor while also can be recovered from lost track due to its search strategy. Obviously, our approach (i.e. AdaCov) has got the lowest pixel error due to its thoughtful balance of both local and global search strategy. Overall, AdaCov has the best tracking accuracy among these approaches.

We investigated the six approaches on “Woman” sequence in more detail. The woman sequence is a typical sequence with some partial occlusions, illumination changes and possible distractions. For local search-based methods, i.e. HistMS, GradientCov and LocalCov, we take a re-initialization. Centroid distance metric is used to further evaluate the tracking accuracy at each frame on “Woman” sequence for different approaches.

Figure 3 shows the normalized average centroid distance for “Woman” sequence. GlobalCov has some frames with large centroid distances due to distractions. It is distracted by some regions in the image which are far from the really objects. But it can catch the object again and need not re-initialization at all. ColorPF frequently loses the target due to its weak descriptor and distraction while it can recover from lost track. HistMS loses its track



**Fig. 3** Normalized average centroid distance for “Woman” sequence

frequently, so we have to re-initialization to keep it can track the really object again. Even we have adopt re-initialization operation (68 times), it still have higher average centroid distance. In fact, HistMS is the worsts due to its overall bad performance while GlobalCov is just a few frames jumping far away due to distraction. LocalCov, GradientCov and AdaCov are better. They are actually gets almost similar results because the whole sequence only has partial occlusion, without full occlusion, but AdaCov is still the best. It also means AdaCov has the smoothest tracking trajectory.

Table 1 further shows the average pixel error and re-initialization number for “Woman” sequence. GlobalCov, ColorPF and AdaCov have the ability to re-locate the object by themselves. Therefore, they need not to set re-initialization operation. ColorPF is the worst in spite of its recover ability. HistMS is the sub-worst, which not only has the highest re-initialization number, but also have the sub-highest average pixel error. LocalCov also need a few re-initialization operations, but it has small average pixel errors. AdaCov has the minimum average pixel error and need not re-initialization operation.

Obviously, AdaCov has the best performance on this sequence. Except for “Face” sequences, we also get similar result with other sequences without full occlusion.

## 4.2 Robustness

For the sequence with full occlusion, we evaluate its lost track value. Table 2 show the lost track number and rate for Jogging sequence.

It can be seen from Table 2 that the lost track rates of the HistMS, GradientCov and LocalCov are very high because they lose the target early from frame 63, and then they never re-locate the object anymore. GlobalCov loses some frame due to the distraction. It loses 14 frames in the procedure. However, GlobalCov can re-locate the object even after losing track due to its global nature. ColorPF loses a lot of frames due to distraction and its weak feature descriptor, although it can recover from lost track sometimes. AdaCov is good at dealing with the lost track problem because it trades off the anti-distraction and re-location abilities.

Figure 4 shows the situations for full occlusion in “Jogging” sequence. The first row is HistMS with frames (46, 62, 66, 87). It loses its track when occlusion occurs and then cannot catch the object anymore. The second row is GlobalCov with frames (46, 52, 56, 57). It also loses its track when occlusion occurs, and is distracted by other objects, but it catches the object again when the occlusion disappears. Therefore, the tracking trajectory of the object is not steady. The third row is LocalCov with the frames (46, 55, 57, 69) and the fourth row is GridentCov with frames (46,55,58,64). they lose their track when occlusion happens and do not capture the target anymore, similar to HistMS. The fifth row is ColorPF with frames (46,49,62,250), it lose its track from frame 62 and then catches the target again

**Table 1** Average Pixel Error and re-initialization number

Tracker vs. dataset	Average pixel error	Re-initialization number
GlobalCov	32.44	0
HistMS	36.76	68
LocalCov	9.49	8
ColorPF	42.32	0
GradientCov	9.42	8
AdaCov	9.00	0

**Table 2** Lost track number and rate

Tracker vs. dataset	Number	Rate
GlobalCov	14	0.045
HistMS	244	0.794
LocalCov	244	0.794
ColorPF	268	0.876
GradientCov	244	0.794
AdaCov	0	0

after a long time. The last row is AdaCov with frames (46, 48, 56, 58), and the bigger yellow window is the enlarged search window. It detects the occlusion situation, then enlarges the search window and keeps the inertia speed for moving, and catches the target again when it appears. Obviously, AdaCov reasonably handles the full occlusion and keeps the tracking trajectory stable. Its enlarged search window makes sure of detecting the target again when occlusion disappears.

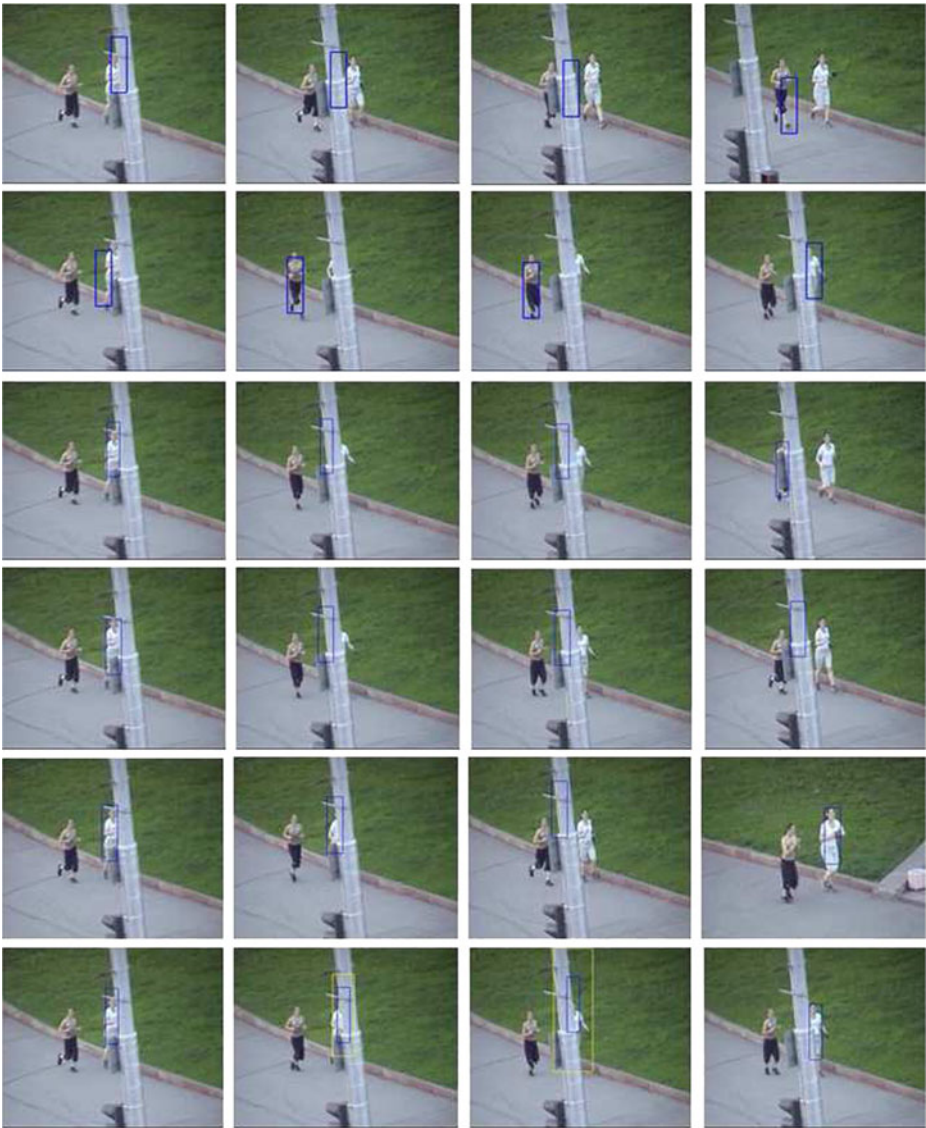
Figure 5 shows the situations for part occlusions, distractions and illumination changes in “Woman” sequence without re-initialization. The first row is HistMS with frames (104,123,126,178,300), which loses the target from frame 123 and never catches it again. The second row is GlobalCov with frames (90,93,101,184,265), which loses the target at frames 93 and 184 while catches them again at frames 101 and 265, respectively. The third row is LocalCov with frames (104,178,211,233,300) and the fourth row is GridentCov with frames (104,178,211,233,300). Both of them lose the track from frame 178 and never catch it again. The fifth row is ColorPF with frames (104,125,250,261,332), which loses the target for a lot frames and catch it again lately. The last row is AdaCov with frames (104,128,149,209,286), and the bigger yellow window is the enlarged search window.

The sudden motions or fast movements are almost similar to the full or partial occlusion situation as shown above. Therefore, our approach also can cope with the sudden motion or fast movement.

### 4.3 Computational efficiency

Computational efficiency is very important to a tracking approach. We implemented our own versions for these six approaches to compare their execution times. The baseline is GlobalCov approach [31] which is used to calculate the speedups obtained by the enhanced approaches. GlobalCov tracker [31] claims to obtain tracking times of about 600 msec per frame for  $320 \times 240$  images using various optimizations including integral images. We define the speedup as the ratio of the execution time of the GlobalCov to that of the given algorithm. Using this information and the relative speedup factors we can estimate the running times per frame for each tracking algorithm. The execution time per  $320 \times 240$  frame is given by  $(600 \div \text{speedup})$  msec.

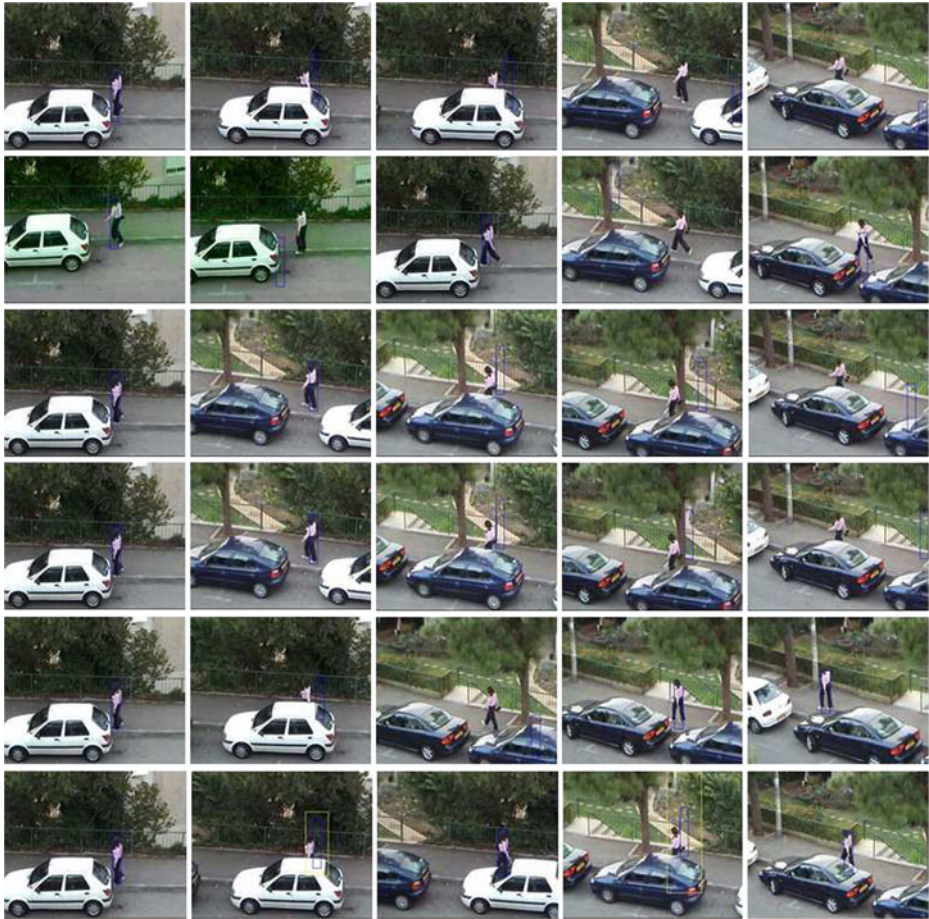
Table 3 reports the speedup factors (relative to the GlobalCov approach) and the estimated execution times per frame obtained by other tracking approaches. The proposed enhancement to the covariance tracker, i.e. AdaCov, results in much faster than GlobalCov while still keeps good accuracy and robustness. Of course, the speedup times is related to the ratio of target region size and image size. Generally, the target region size is much smaller than the whole image (most of them are about 1/40 of image size). Therefore, the speedup is significant.



**Fig. 4** Full occlusion in Jogging sequence. The *first row* is HistMS with frames (46, 62, 66, 87); The *second row* is GlobalCov with frames (46, 52, 56, 57); The *third row* is LocalCov with frames (46, 55, 57, 69); the *fourth row* is GradientCov with frames (46,55,58,64); the *fifth row* is ColorPF with frames (46,49,62,250); The *last row* is AdaCov with frames (46, 48, 56, 58), and the *bigger yellow window* is the enlarged search window

It can be seen from Table 3 that the area of target region is about  $1/45.56$  of the area of the whole image. HistMS gets the best speedup, but it is totally different from the other three covariance-based trackers. LocaCov also gets a good speedup by decreasing the search range. GradientCov is fast due to optimizing by gradient ascent technique and local search strategy. AdaCov is not the fastest, but it also gets a great improvement due to its smaller





**Fig. 5** Tracking Pictures in Woman sequence. The *first row* is HistMS with frames (104,123,126,178,300); The *second row* is GlobalCov with frames (90,93,101,184,265); The *third row* is LocalCov with frames (104,178,211,233,300); the *fourth row* is GridentCov with frames (104,278,211,233,300); the *fifth row* is ColorPF with frames (104,125,250,261,332); The *last row* is AdaCov with frames (104,128,149,209,286), and the *bigger yellow window* is the enlarged search window

search window and fast integral region computation, which is approximate to real-time video processing.

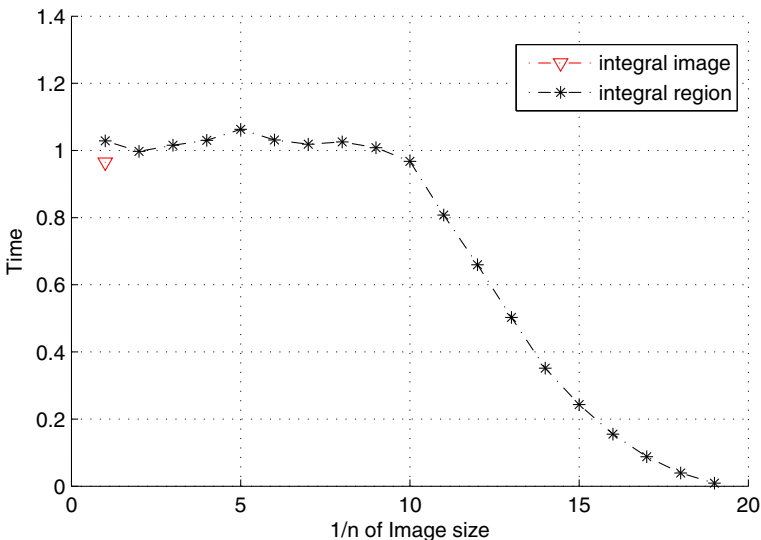
For AdaCov, the construction of integral region instead of integral image helps a lot for the speedup. Assume the target region is  $1/n$  of the whole image, we can see that the time of integral region construction decreases dramatically with bigger  $n$ . Generally, the object region is far smaller than whole image, therefore it is very helpful to decrease the computation time. Of course, when the target region is very large, the time of integral region construction will approach to that of the integral image. In fact, integral region will regress into integral image when the region becomes bigger and bigger. Figure 6 shows the relative construction time of the integral image and the different size of integral region.

There are  $W \times H$  times to compute the  $P(x', y', i)$  and  $Q(x', y', i, j)$  for Integral Image while  $(W_2 + W_1) \times (H_2 + H_1) = (s_w + 1)W_1 \times (s_H + 1)H_1$  times for Integral Region. Meanwhile, there are  $(W - W_1) \times (H - H_1)$  times to search in the whole image while

**Table 3** Tracking time comparison (unit: ms/frame). The jogging sequences with 352\*288 image size, and object region size is about 1/45.56 of image size

Tracker	Speedup (times)	Execution time per frame (msec)
GlobalCov	1	792
HistMS	343.433	2.306
LocalCov	13.862	57.135
ColorPF	75.242	10.526
GradientCov	88.531	8.946
AdaCov	17.253	45.905

$W_2 \times H_2 = s_W \cdot W_1 \times s_H \cdot H_1$  times to search in the search region (i.e. search window). As we said before,  $s_W = 1$  and  $s_H = 1$  in the normal steady state, and  $s_W$  and  $s_H$  will be larger when there are occlusions possibly detected by the occlusion detector. Therefore, there is a remarkable speedup when the size of target window is much smaller than that of the whole image. For Example, the size of the white girl in Jogging sequence is about 1/45 (about 1/3 of height and 1/15 of width) to that of the whole image. therefore, the speedup of the construction of Integral is about  $\frac{W \times H}{(W_2+W_1) \times (H_2+H_1)} = \frac{45}{(s_W+1) \times (s_H+1)}$  while the speedup of the searching is about  $\frac{(W-W_1) \times (H-H_1)}{W_2 \times H_2} = \frac{28}{s_H \cdot s_W}$ . In normal steady state, i.e.  $s_W = 1$  and  $s_H = 1$ , the speedups are 11.25 and 28 relatively for Integral and searching. In occlusion condition (assume  $s_W = 2$  and  $s_H = 2$ ), there are also speedups of 6 and 7 for Integral and searching relatively. In most of cases, the tracking target is much smaller than the whole image and the tracking process is in normal condition (without occlusion). Therefore, the proposed approach is valuable for speedup.



**Fig. 6** The relative time of integral region construction for different target region (1/n of image size)



#### 4.4 Discussion

From the previous experiments, we can see that: HistMS is the fastest, but it has the worst accuracy (e.g. maximum average centroid distance) and robustness (e.g. maximum re-initialization number and lost track) due to its low discriminative ability of feature and fixed local range; GlobalCov has good robustness (e.g. it can handling full occlusion and need not re-initialization) and normal accuracy, but it is easy to be affected by distractions and has the slowest processing time due to its global nature and brute-force strategy; LocalCov and GradientCov have better accuracy in steady states and faster processing time, but it is not robust for occlusion and need re-initialization due to their local mode. ColorPF's accuracy is not very good, but its speed is promising and it has recover ability. AdaCov has the best accuracy, computational efficiency and robustness, because it makes full use of the advantage of the global and local mode and has discriminative feature descriptor. It is adaptive to different situation by occlusion detection. AdaCov is more like to LocalCov and GradientCov when the tracking is in steady states, while it regresses into GlobalCov gradually when the tracking is in abnormal situations.

Our approach is based on the traditional covariance tracking (i.e. GlobalCov in the manuscript). It inherits the merits of GlobalCov, i.e. solving the occlusion and sudden motion problems. Meanwhile, it also gets the advantage of the Local covariance tracking (i.e. LocalCov), e.g. the fast speed, less distraction, more stable tracking trajectory. It smoothly switches between these two by a simple occlusion detector. Moreover, AdaCov gets a good tradeoff between the accuracy and speed.

#### 5 Conclusion

We have described a novel generalized, adaptive covariance tracking approach named AdaCov in the pursuit of accurate, robust and efficient tracking. The proposed approach takes the advantages of local and global tracking strategies and switches smoothly by occlusion detection. The proposed tracking algorithm is efficient because local searching strategy is adopted for most of the frames. It can deal with occlusions and large motions by switching from local to global matching. Moreover, the integral region generalizes the integral image and is adaptive to different regions with much faster computation. We will further improve the accuracy of occlusion detector and try to put the AdaCov into particle filtering tracking framework in near future.

**Acknowledgments** This work is supported by National Natural Science Foundation of China (No.61170093) and China Postdoctoral Science Foundation (No. 20110491149).

#### References

1. Adam A, Rivlin E, Shimshoni I (2006) Robust fragments-based tracking using the integral histogram. In: Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2006)
2. Andriyenko A, Schindler K, Roth S (2012) Discrete-continuous optimization for multi-target tracking. In: Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)
3. Arsigny V, Fillard P, Pennec X, Ayache N (2006) Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magnet Reson Med* 56:411–421
4. Arsigny V, Fillard P, Pennec X, Ayache N (2007) Geometric means in a novel vector space structure on symmetric positive definite matrices. *SIAM J Matrix Anal Appl* 29:328–347

5. Babenko B, Yang MH, Belongie S (2009) Visual tracking with online multiple instance learning. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2009)*
6. Bao CL, Wu Y, Ling HB, Ji H (2012) Real time robust L1 tracker using accelerated proximal gradient approach. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
7. Cherian A, Sra A, Banerjee A, Papanikolopoulos N (2011) Efficient similarity search for covariance matrices via the Jensen-Bregman LogDet divergence. In: *Proceeding of the 13th international conference on computer vision (ICCV 2011)*
8. Collins RT (2003) Mean-shift blob tracking through scale space. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2003)*
9. Comaniciu D, Meer P (2002) Mean shift: a robust approach toward feature space analysis. *IEEE Trans Pattern Anal Mach Intell* 24:603–609
10. Comaniciu D, Ramesh V, Meer P (2000) Real-time tracking of non-rigid objects using mean shift. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2006)*
11. Comaniciu D, Ramesh V, Meer P (2003) Kernel-based object tracking. *IEEE Trans Pattern Anal Mach Intell* 25:564–577
12. Duan G, Ai H, Cao S, Lao S (2012) Group tracking: exploring mutual relations for multiple object tracking. In: *Proceeding of the 12th European conference on computer vision (ECCV 2012)*
13. Grabner H, Leistner C, Bischof H (2008) Semi-supervised on-line boosting for robust tracking. In: *Proceeding of the 10-th European conference on computer vision (ECCV 2008)*
14. Hare S, Saffari A, Torr P (2011) Struck: structured output tracking with kernels. In: *Proceeding of the 13th international conference on computer vision (ICCV 2011)*
15. Hare S, Saffari A, Torr PHS (2012) Efficient online structured output learning for keypoint-based object tracking. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
16. Jia X, Lu H, Yang MH (2012) Visual tracking via adaptive structural local sparse appearance model. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
17. Jiang N, Liu W, Wu Y (2012) Order determination and sparsity-regularized metric learning adaptive visual tracking. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
18. Kalal Z, Mikolajczyk K, Matas J (2010) Face-TLD: tracking-learning-detection applied to face. In: *Proceeding of the International Conference on Image Processing (ICIP 2010)*
19. Kalal Z, Matas J, Mikolajczyk K (2010) P-N learning: bootstrapping binary classifiers by structural constraints. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2010)*
20. Kwon J, Lee KM (2010) Visual tracking decomposition. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2010)*
21. Li X, Shen C, Shi Q, Dick A, Hengel AVD (2012) Non-sparse linear representations for visual tracking with online reservoir metric learning. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
22. Liu J, Carr P, Collins RT, Liu Y (2013) Tracking sports players with context-conditioned motion models. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2013)*
23. Maggio E, Cavallaro A (2005) Multi-Part target representation for color tracking. In: *Proceeding of the IEEE international conference on image processing (ICIP 2005)*
24. Mei X, Ling H (2009) Robust visual tracking using L1 minimization. In: *Proceeding of the IEEE 12th international conference on computer vision (ICCV 2009)*
25. Nguyen QA, Robles-Kelly A, Shen C (2007) Kernel-based tracking from a probabilistic viewpoint. In: *Proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR 2007)*
26. Parameswaran V, Ramesh V, Zoghiani I (2006) Tunable Kernels for tracking. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2006)*
27. Park DW, Kwon J, Lee KM (2012) Robust visual tracking using autoregressive hidden Markov model. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
28. Pennec X, Fillard P, Ayache N (2006) A riemannian framework for tensor computing. *Int J Comput Vis* 66:41–66
29. Perez P, Hue C, Vermaak J, Gangnet M (2002) Color-based probabilistic tracking. In: *Proceeding of the 7th European conference on computer vision (ECCV 2002)*
30. Porikli F, Tuzel O (2005) Multi-Kernel object tracking. In: *Proceeding of the IEEE international conference on multimedia and expo (ICME 2005)*
31. Porikli F, Tuzel O, Meer P (2006) Covariance tracking using model update based on lie algebra. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2006)*

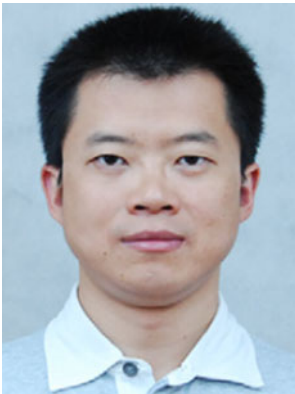
32. Ross DA, Lim J, Lin RS, Yang MH (2008) Incremental learning for robust visual tracking. *Int J Comput Vis* 77:125–141
33. Rui Y, Shi Q, Shen S, Zhang Y, Hengel AVD (2013) Part-based visual tracking with online latent structural learning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2013)*
34. Tuzel O, Porikli F, Meer P (2006) Region covariance: a fast descriptor for detection and classification. In: *Proceeding of the 9th European conference on computer vision (ECCV 2006)*
35. Tyagi A, James WD, Potamianos G (2008) Steepest descent for efficient covariance tracking. In: *Proceeding of the IEEE workshop on in motion and video computing (WMVC 2008)*
36. Viola P, Jones MJ (2004) Robust real-time face detection. *Proc Int J Comput Vis* 57(2):137–154
37. Wang J, Yagi Y (2008) Switching local and covariance matching for efficient object tracking. In: *Proceeding of the international conference on pattern recognition (ICPR 2008)*
38. Wu Y, Cheng J, Wang J, Lu H (2009) Real-time visual tracking via incremental covariance tensor learning. In: *Proceeding of the 12th international conference on computer vision (ICCV 2009)*
39. Wu Y, Cheng J, Wang J (2012) Real-time probabilistic covariance tracking with efficient model update. *IEEE Trans Image Process* 21(5):2824–2837
40. Wu Y, Lim J, Yang MH (2013) Online object tracking: a benchmark. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2013)*
41. Xu D, Wang Y, An J (2005) Applying a new spatial color histogram in mean-shift based tracking algorithm. In: *Proceeding of the New Zealand conference on image and vision computing*
42. Xu J, Lu H, Yang MH (2012) Visual tracking via adaptive structural local sparse appearance model. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
43. Yang B, Nevatia R (2012) An online learned CRF model for multi-target tracking. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
44. Yang B, Nevatia R (2012) Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
45. Yilmaz A, Javed O, Shah M (2006) Object tracking: a survey. *ACM Comput Surv* 38:1–45
46. Zamir AR, Dehghan A, Shah M (2012) GMCP-Tracker: global multi-object tracking using generalized minimum clique graphs. In: *Proceeding of the 12th European conference on computer vision (ECCV 2012)*
47. Zhang L, Maaten LVD (2013) Structure preserving object tracking. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2013)*
48. Zhang T, Ghanem B, Ahuja N (2012) Robust visual tracking via multi-task sparse learning. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*
49. Zhong W, Lu H, Yang MH (2012) Robust object tracking via sparsity-based collaborative model. In: *Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR 2012)*



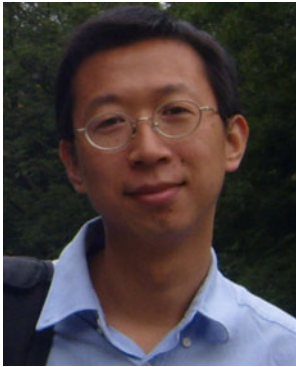
**Ruhan He**, is an Associate Professor in College of Mathematics and Computer Science, Wuhan Textile University, China. He received his MSc and PhD degree from Huazhong University of Science and Technology (HUST), in 2004 and 2007, respectively. His research focuses on Machine Learning, Multimedia Processing and Analysis, etc. He has authored more than thirty scientific papers and articles. He also initialized and led over multiple research projects.



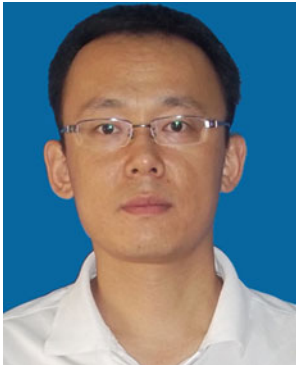
**Bing Yang**, is an Associate Professor of computer science at HuBei University. He received the BS degree from Central China Normal University, Wuhan, China. He received the MS and PhD degrees in computer science from Huazhong University of Science and Technology in 2004 and 2008, respectively. His research interests include Multimedia, Mobile Computing and Wireless Networks.



**Nong Sang**, is currently a professor at the Institute for Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, Wuhan, China. He graduated from Huazhong University of Science and Technology and received his BE degree in computer science and engineering in 1990, MS degree in pattern recognition and intelligent control in 1993, and PhD degree in pattern recognition and intelligent systems in 2000. His research interests include computationally modeling of biological vision perception and applications in computer vision, image analysis and object recognition based on statistical learning, medical image processing and analysis, interpretation of remote sensing images, and intelligent video surveillance.



**Yongsheng Yu**, is currently a researcher in The Green Building Materials and Manufacturing Engineering Research Center of Ministry of Education at Wuhan University of Technology. He received his B.S. degree in Computer Science in Wuhan University of Hydraulic and Electrical Engineering in 2001. He obtained his M.S. and Ph.D. degrees in Computer Science from Huazhong University of Science and Technology in 2005 and 2010, respectively. His research interests include computer security, image processing and computer simulation.



**Geli Bai**, is an Associate Professor in College of Computer and Information Engineering, Inner Mongolia Agricultural University, China. He received his MSc degree from Inner Mongolia Normal University, in 2008. His research focuses on Machine Learning, Operating System, etc.



**Jizi Li**, is a Professor at Supply Chain System Research Center, Wuhan Textile University, Wuhan, China. He received his PhD and Master degrees in Huazhong Agriculture University, and Wuhan University of Technology, respectively. His research interests include Networks Simulation and Optimization Theory. Some of his works were published in International Journal of Production Economics, Nonlinear Analysis, Abstract and Applied Analysis, Acta Mathematicae Applicatae Sinica.