



# Robust visual tracking based on adaptive gradient descent optimization of a cost function with parametric models of appearance and geometry

Younes Dhassi<sup>1</sup> · Abdellah Aarab<sup>1</sup>

Received: 30 May 2018 / Revised: 3 January 2019 / Accepted: 18 February 2019 /

Published online: 18 March 2019

© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

In the field of visual tracking, there are many issues to consider which make the development of a robust tracking method very difficult, among these complications; the appearance change of the target, the fast motion, the background clutter, the camera motion, scale variation and the in plane Rotation. To override these problems, we develop an effective general framework for object tracking that addresses most of these issues. First the tracking problem is formulated in the form of a robust cost function which is a composition of the appearance and dynamic model, this formulation ensures the integration of the appearance and motion informations. Second the minimization is accomplished by the gradient descent optimization with adaptive step size prediction, the step size adaptation accelerates the optimization process and increases the accuracy. Throughout, we present experimental results made on different challenging sequences, the experimentations results demonstrate the efficiency and effectiveness of our methods.

**Keywords** Visual tracking · Gaussian mixture model · Expectation maximization · Gradient descent

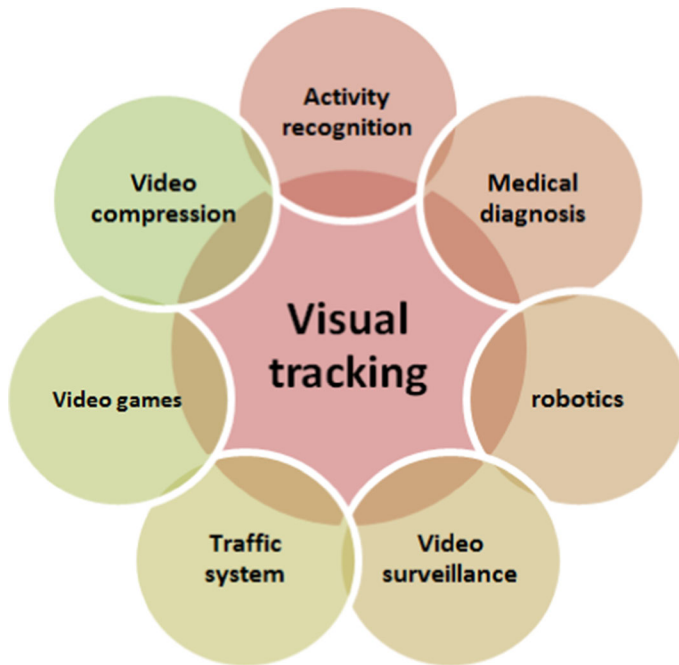
## 1 Introduction

Object tracking is a common problem in the field of computer vision. The constant increase in the power of computers, the reduction in the cost of cameras and the increased need for video analysis have engendered a keen interest in object tracking algorithms [16]. This type of treatment is today at the center of many applications Fig. 1, in smart visual surveillance [21, 29], unmanned vehicles [25]. Visual tracking is applied to estimate a set of parameters related to appearance and motion, these parameters are collected and processed to understand the

---

✉ Younes Dhassi  
dyounes2003@gmail.com

<sup>1</sup> Laboratory of Electronics, Signals, Systems and Computers, Department Of Physics Faculty of Sciences Dhar- Mahraz, Sidi Mohamed Ben Abdellah University, Fes, Morocco



**Fig. 1** Some applications of visual tracking

behavior of the target, hence the visual tracking is potentially applied in human computer interaction [6, 19], intelligent robot [27] and Intelligent traffic system [4]. Other fields whose application of visual tracking is promising, especially in the case of modern medicine [24], for example speckle-tracking echocardiography improves the detection of myocardial infarction over visual assessment of systolic wall motion abnormalities. In fact, it becomes a novel technological tool in medical clinical diagnostics and therapeutics. It can be applied to improve health status and exercise habits by a healthcare intelligent computer.

The tracking corresponds to the estimation of the location of the object in each of the images of a video sequence, the camera and / or the object being able to be simultaneously in motion. The localization process is based on the recognition of the object of interest from a set of visual characteristics such as color, shape, speed, etc. As shown in Fig. 2, the structure of the majority of the visual tracking method consists of a set of components namely: Appearance modeling; Motion modeling; Estimation; Extraction and model update.

There are many challenging issues Fig. 3, which make the development of a tracking method [1] very difficult. This difficulty arises from several conditions, namely variation in target appearance, pose or target deformations large scale and orientation changes. Therefore failing to take account of one of these two challenges can lead to a weak tracking process.

This paper aims at developing a robust tracker which runs in several difficult situations; indeed the tracked object can undergo radical changes due to the geometric transformations of the object, as well as changes in illumination. In this context we propose a new tracking algorithm Fig. 4, which addresses each of these complications. The visual and the motion information are integrated and an adaptive optimization is performed to estimate the state of the target in the next frame. The problem is formulated on a robust cost function which is a composition of the appearance and motion model. First the target's appearance is modeled using the Gaussian mixture model (GMM), which is more discriminative and less massive than a description based on color or oriented gradient histograms.

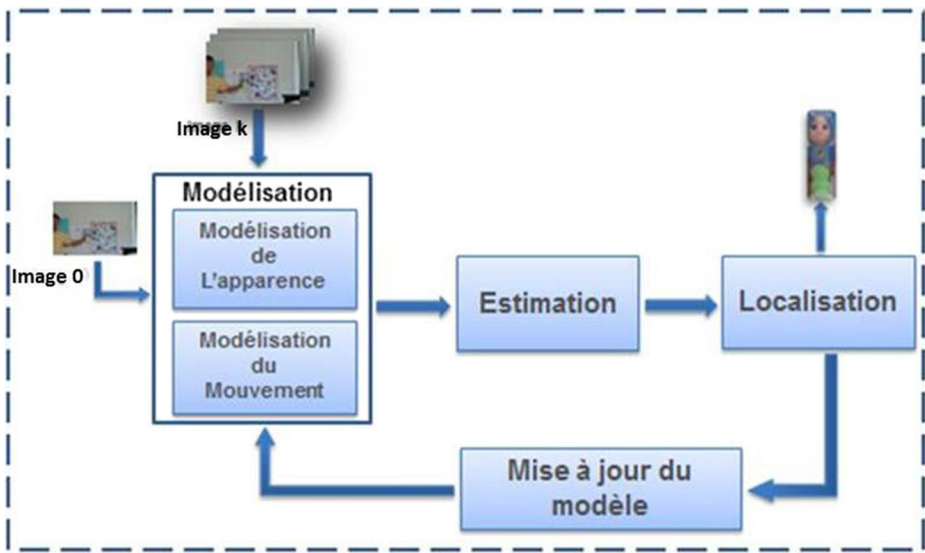


Fig. 2 Flowchart of visual tracking



Fig. 3 Different problems that can occur during the tracking. **a** occlusion; **b** illumination variation; **c** cluttered background; **d** scaling; **e** deformation; **f** fast motion; **g** Rotation; **h** motion blur

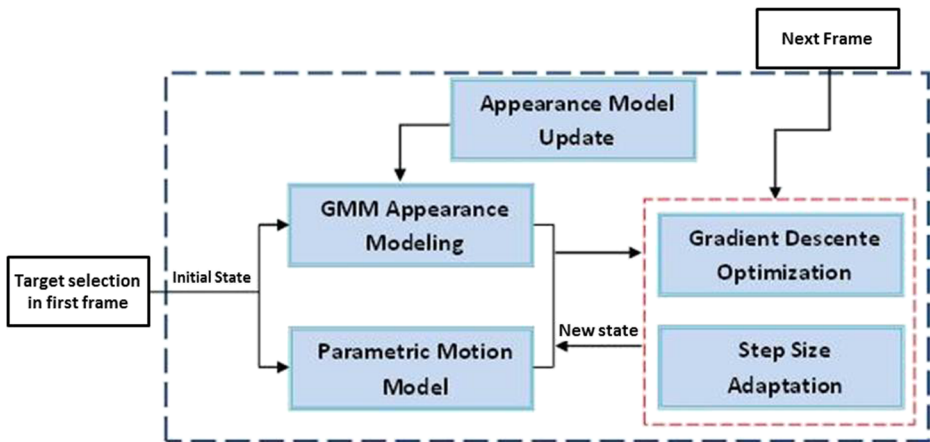


Fig. 4 Flowchart of the proposed visual tracking algorithm

Second the target's motion is modeled using a parametric motion model which integrates the main geometric transformations, then both models are integrated in a single objective function. Finally the minimization is accomplished by an improved gradient descent optimization with adaptive step size prediction, the step size adaptation accelerates the optimization process, and therefore the tracking can run in real time.

There are two main contributions in the proposed work. 1) A robust minimization model formulated by a cost function integrating homogeneously the appearance model on the one hand, and on the other hand a parameterized motion model which describes the displacements of the target and which takes in consideration of basic transformations namely translation, scale and orientation changes. 2) The introduction of a fast numerical optimization method to solve the minimization problem, the speed of the minimization process is ensured by the use of the gradient descent based on an adaptive decaying step size strategy. The proposed adaptive step size produces a sufficient reduction with a favorable descent direction, which allows a fast and an effective exploration in the target area specially in case of large displacement and scale as well as when the target changes its orientation. Therefore our tracking algorithm can converge quickly, so that makes our method usable in real time application.

The remainder of the paper is organized as follows: The most related works are summarized in Section 2. Section 3 presents the target appearance model. In Section 4 demonstrates details of the proposed method. Experiments results and performance discussion are reported in Section 5. Finally, the paper is concluded in Section 6.

## 2 Related work

In last decade many visual tracking methods have been proposed, which can be categorized in two classes generative and discriminative. The generative methods represent the object by appearance model that can be obviously convoluted by a kernel, then the tracking process seeks the candidate whose observed appearance model is most similar to that of the template, among the popular generative methods one can quote, kernel-based object tracking [3], particle filter [18, 32], Particle-Kalman Filter [8], online appearance models for visual tracking [28, 30]. The discriminative methods try to separate the tracked object from the background by a binary classifier, the

most widespread discriminative methods include online selection of discriminative tracking features [5], object tracking using incremental 2d-lda learning and Bayes inference [15], ensemble tracking [2]. Thereafter we will be limited to briefly presenting the most related methods to our own, for more details the reader can refer to a detailed review in [17]. In [14] multiple reference histograms obtained from multiple available prior views of the target are adopted as the appearance model, the authors propose an extension to the mean shift tracker, where the convex hull of these histograms is used as the target model. The proposed method is stable to appearance change but not enough motion information taken into account in the formulation of the problem, which can weaken the tracking in the case of complex motion. The authors in [31] represent the appearance model of the template and the candidate using background weighted histogram and color weighted histogram, the proposed method termed as adaptive pyramid mean shift uses pyramid analysis with adaptive levels and scales for better stability and robustness. The proposed method can accurately track the object in the cluttered and in variational scene but it can show weaknesses in the case of changes of scale of orientation or complex motion. In [9] a novel tracking algorithm is proposed based on combination of mean shift tracker with the online learning-based detector, the proposed algorithm can reinitialize the target when it converges to a local minima and it can cope with scale changes, occlusions and appearance changes. In order to ensure long-term tracking the target model is updated. In addition, to make the tracker real time operating, the Kalman filter and the Mahalanobis distance is used to obtain the validation region. The algorithm is effective to track targets in complex environments that contain full occlusions or object reappearances. However the tracker predicts the target position based on the constant velocity model. The motion model used is weak and non-adaptive. Indeed, if the target does not follow the constant velocity model, it is difficult to predict the accurate target position and the validation region is defined wrongly. In [11] a probabilistic real time tracking algorithm is proposed the target's appearance model is represented by a Gaussian mixture model, the tracking is achieved by maximizing its weighted likelihood in the image sequence using the gradient descent optimization, withal, the formulation of the cost function does not integrate any dynamic model. The proposed method handles scale and rotation changes of the target, as well as appearance changes, but when the motion is large due to the fact that the camera or the target move quickly and no overlapping section exists between the target in two consecutive frames, the tracker fails. To overcome the problems encountered in the case of 3-D human motion tracking namely the inherent silhouette-pose ambiguities and the various motion types, the authors in [6] propose a fusion method which integrates low- and high-dimensional approaches into one framework. The proposed strategy has the advantage of allowing the two trackers to cooperate and complement each other in order to resolve the problems mentioned before. The proposed algorithm is capable of performing robustly with various styles of motion and with different camera views. The authors in [22] use the sparse representation to model the Target. The tracking problem is formulated as a L1 norm related minimization problem, the optimization of the cost function is resolved based on Augmented Lagrange Multiplier method, indeed the augmented cost function is complex and no motion information are present in the formulation of it. The experiments validate the tracking accuracy and time efficiency of the proposed tracker on challenging conditions namely large illumination variation and pose change. However performance decreases in the case of sequences that present large scale and fast motion. In [13] the authors propose the  $\ell_p$ -regularization collaborative appearance model and the minimization problem of  $\ell_p$ -regularization are solved using the Accelerated Proximal Gradient approach applying the Generalization of Soft-threshold operator. The proposed method can achieve more favorable performance in case of drastic illumination change and Background Clutter, but in case

of scale change, rotation, and blur as well in the case of large displacement the tracker shows weaknesses. Finally in [7], the authors consider the overall dynamical behavior of the object as being linear, thus modeling the motion in a global view by two parametric single acceleration model, the filter  $H_\infty$  is used to estimate the state related to each of the two sub models. The estimates are then considered as an input for the particle filter to compute the local estimate for each of the two inputs, and finally the interactive multiple model algorithm mix the local estimates according to their Model probability to provide information about the posterior location of the target. The proposed method handles many challenging situations affectively, but the computation time is not encouraging enough for it to be applied in real-time scenarios.

On the whole, the methods presented above models indeed offer flexibility for the tracker to cope with the change of the target’s appearance and background clutter, yet the formulation of the cost function for the tracking problem remains fairly limited, on the one hand the motion information is presented in a restricted way seen even absent, on the other hand the optimization process is more computationally expensive owing to the complexity of the objective function and the optimization algorithm.

### 3 Target appearance modeling

The target area is defined by a rectangular region  $\Theta$ ; pixel colors in the rectangle area are used to construct the GMM employing the EM algorithm [20]. Let  $p_c$  be a vector representing the coordinates of the center of the target’s area. The coordinates of the  $i$ th pixel in the target area are represented by  $p_{i=1 \dots N} = [x_i \ y_i]$  and the corresponding feature by  $I_i$ . The log-likelihood function for the target’s area  $\Theta_{p_c}$  centered at  $p_c$  is defined by:

$$L(p_c; \pi, m, \Sigma) = \sum_{p_i \in \Theta_{p_c}} \ln \sum_{k=1}^K \pi_k N(I_i, m_k, \Sigma_k) \tag{1}$$

The EM algorithm is employed in two step to maximize the log of the likelihood function Eq. 1 with respect to  $\pi$ ,  $m$  and  $\Sigma$ .

In the E-step, the expectations  $z_k(I_i)$  are computed:

$$z_k(I_i) = \frac{\pi_k N(I_i, m_k, \Sigma_k)}{\sum_{k=1}^K \pi_k N(I_i, m_k, \Sigma_k)} \tag{2}$$

In the M-Step, the parameters of the GMM,  $\pi$ ,  $m$  and  $\Sigma$  are estimated as follow:

$$N_k = \sum_{p_i \in \Theta_{p_c}} z_k(I_i) \tag{3}$$

$$m_k = \frac{1}{N_k} \sum_{p_i \in \Theta_{p_c}} z_k(I_i) I_i \tag{4}$$

$$\Sigma_k = \frac{1}{N_k} \sum_{p_i \in \Theta_{p_c}} z_k(I_i) (I_i - m_k)(I_i - m_k)^t \tag{5}$$

$$\pi_k = \frac{N_k}{N} \tag{6}$$

During the tracking process, the appearance of the target may change due to different environmental conditions, which can weaken the model; therefore tracking of the target may

deviate. To solve this problem, the main idea is to allow the model to adapt to different changes in the environment, so dynamically updating the target’s appearance model is essential. For this purpose, new components are inserted into the mixture model using near-target pixels that have low probability. In addition, if the importance of a component becomes low enough, the component is removed from the mixture.

At a certain frequency, with each number of frames  $M$ . We set  $M$  equal to 50, therefore the update will take place every 2 s for 25 frames per second. The new component is initialized with parameters calculated from the pixels that represent the lower quantile of likelihood and with a low weight. Subsequently the EM algorithm is used to estimate the mean and the correct covariance of the new component. The parameters of the basic mixture model, the mean and the covariance, are set to avoid the drift problem, only their mixing proportions change due to the insertion of the new component. Finally, if the importance of a component is less than a threshold (less than  $0.1 / K$ ), the component is removed from the mixture.

### 4 Tracking by adaptive gradient descent optimization

Subsequently we present the details of our approach for visual tracking under the conditions previously explained. The GMM-based appearance model and the parameterized motion model are merged; hence the tracking problem is formulated by a single objective function. Thus the gradient method is used to optimize the parameters of the model; the descent step is adaptively updated at each iteration according to logical rules. The flowchart of the proposed optimization algorithm is present by Fig. 5, and the pseudo code is given by Algorithm. 1.

The objective is to seek in the next frame for the candidate’s area  $I$  centered at  $c$  position, whose pixels realize a maximum log-likelihood function; therefore the objective function to maximize is given as:

$$L(T(p_c, \mu); \pi, m, \Sigma) = \sum_{p_i \in T(\Theta_{p_c, \mu})} \ln \sum_{k=1}^K \pi_k N(I_i, m_k, \Sigma_k) \tag{7}$$

In which  $T(x, \mu)$  is a parametric motion model parameterized by  $\mu = (u_x, u_y, s_x, s_y, \theta)^t$ , the linear motion models is expressed as:

$$T(x, \mu) = sR(\theta)x + u \tag{8}$$

In which

$$R(\theta) = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \quad s = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \quad u = [u_x \quad u_y]^t$$

And  $I(T(p, \mu))$  the candidate’s region under the change of coordinates with parameters  $\mu$

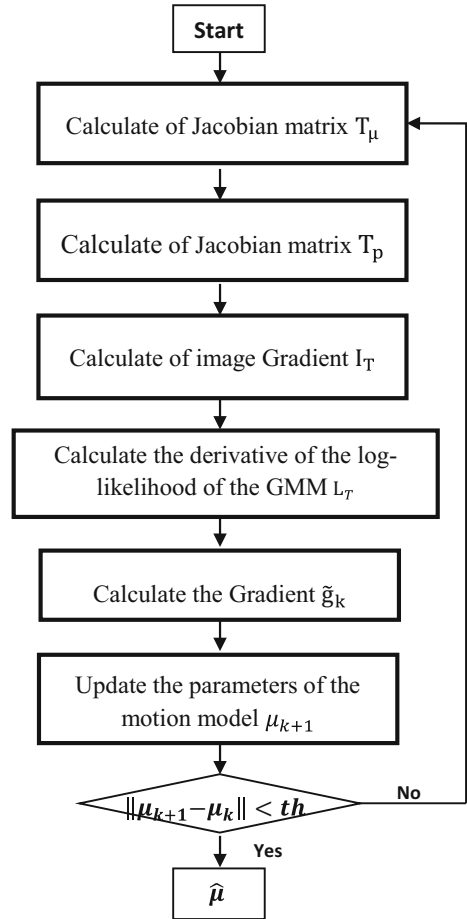
Thereafter we use the gradient descent method to solve the optimization tracking problem, formulated as follows:

$$\hat{\mu} = \arg \min L(T(p, \mu); \pi, m, \Sigma) \tag{9}$$

The gradient descent method adopts iterative updates to obtain the optimal parameters using the following form:

$$\mu_{k+1} = \mu_k - \alpha_k \tilde{g}_k \tag{10}$$

**Fig. 5** Flowchart of the adaptive gradient descent optimization



Where  $\alpha_k$ , is the step size at iteration k and  $\tilde{g}_k$  is the gradient of the cost function

$$\tilde{g}_k = \frac{\partial \varphi(\mu)}{\partial \mu_k} \tag{11}$$

### 4.1 Determination of gradient

By applying the derivative chain rules to equation the following gradient expressions are obtained:

$$\tilde{g}_k = L_T T_\mu \tag{12}$$

Where  $L_T = \frac{\partial L(T(p, \mu); \pi, m, \Sigma)}{\partial \mu}$  and  $T_\mu = \frac{\partial T(p, \mu)}{\partial \mu}$

In which  $L_T$  is the gradient of L with respect to the components of the vector T.  $T_\mu$  is the Jacobian matrix of the transformation T function of  $\mu$ .

Recall that the Jacobian matrix of the transformation T function of  $\mu$  is the  $2 \times 5$  matrix:

$$T_\mu = \left( \frac{\partial T(p, \mu)}{\partial u_x} \mid \frac{\partial T(p, \mu)}{\partial u_y} \mid \frac{\partial T(p, \mu)}{\partial s_x} \mid \frac{\partial T(p, \mu)}{\partial s_y} \mid \frac{\partial T(p, \mu)}{\partial \theta} \right) \tag{13}$$



Where  $T_p$  is the  $2 \times 2$  Jacobian matrix of  $T$  treated as a function of  $p = (p_x, p_y)^t$

---

**Algorithm 1** Pseudo algorithm of the proposed method

---

1. Selection the object in the first frame
  2. Run the EM algorithm to build the GMM appearance model
  3. Initialize the parameters of the motion model
  4. Initialize the descent step to a great value
  5. **for**  $k \leftarrow 1$  **to**  $Nframe$ , **do**
  6. Calculate of Jacobian matrix  $T_\mu$  using Eq. 13
  7. Calculate of Jacobian matrix  $T_p$  using Eq. 14
  8. Calculate of image Gradient  $I_T$  using Eq. 17
  9. Calculate the derivative of the log-likelihood of the GMM  $L_T$  using Eq. 15
  10. Calculate the Gradient  $\tilde{g}_k$  using Eq. 12
  11. Update the descent step size using the rules explained in section 4.2
  12. Update the parameters of the motion model  $\mu_{k+1}$  using Eq.10
  13. Update de appearance model as explained in section 3
  14. **end for**
- 

$$T_p = \left( \frac{\partial T(p, \mu)}{\partial p_x} \mid \frac{\partial T(p, \mu)}{\partial p_y} \right) \tag{14}$$

And  $L_T$  is the derivation of  $L$  with respect to the components of the vector  $T$ , given by:

$$L_T = \sum_{p_i \in I(T(p, \mu))} - \frac{\sum_{k=1}^K \pi_k (I(T(p, \mu)) - m_k)^t \sum_k^{-1} N(I(T(p, \mu)), m_k, \Sigma_k)}{\sum_{k=1}^K \pi_k N(I(T(p, \mu)), m_k, \Sigma_k)} I_T T_\mu \tag{15}$$

In which  $I_T$  is derivation of  $I$  with respect to the components of the vector  $T$ , by applying the chaine rule to  $I_p = \frac{\partial I(T(p, \mu))}{\partial p}$  we obtain

$$I_p = I_T T_p \tag{16}$$

Therefore

$$I_T = I_p T_p^{-1} \tag{17}$$

**4.2 Step size adaptation**

A sequence of appropriately sized steps is very important, a poor choice of the step size negatively influences the convergence, and therefore an adaptive and robust estimate of the step size is essential for efficient and fast gradient descent optimization.

In this paper we propose to use an exponential decay function a function defined as follows:

$$\alpha_k = A \exp^{-\lambda_k k} \tag{18}$$

Then, where  $A > 0$ , it is important for the beginning of optimization, it gives the starting point.  $\lambda \geq 1$  is so important it controls the degree of decaying and determines the global shape of the

step size sequence. The determination of the values of  $A$  and  $\lambda$  is paramount but not obvious, in the following; we develop an automatic approach for the selection of the decay function parameters so as to achieve rapid optimization convergence. The displacement of a pixel which belongs to the tracked area  $I_i$  must follow a decaying scheme; it must start with a large reasonable value and gradually tends to zero. The pixel displacement between iteration  $k$  and  $k + 1$  is defined as:

$$\Delta_k(p) = T(p, \mu_{k+1}) - T(p, \mu_k), \forall p \in I_k \tag{19}$$

The step size selection must be made in such a way that the displacement between each two iterations is neither too large nor too small. Develops in the first-order Taylor expansion the displacement around  $\mu_k$ :

$$\Delta_k(p) \approx T_\mu (\mu_{k+1} - \mu_k) \tag{20}$$

In which  $T_\mu = \frac{\partial T(p, \mu)}{\partial \mu}$  is the Jacobian matrix of the transformation function, using Eq. (10), the Eq. (20) can be rewritten as:

$$\Delta_k(p) \approx -\alpha_k T_\mu \tilde{g}_k \tag{21}$$

According to [12], the magnitude of the displacement  $\|\Delta_k(p)\|$  for each pixel between two iterations should be no larger than  $\delta$  and by using a weakened form for this assumption:

$$P(\|\Delta_k(p)\| > \delta) < 0.05, \forall p \in I \tag{22}$$

According to [26], the Eq. (22) can be approximated by:

$$\alpha_k \left( E \|T_\mu \tilde{g}_k\| + 2\sqrt{\text{Var} \|T_\mu \tilde{g}_k\|} \right) \leq \delta \tag{23}$$

The largest step size is taken at the beginning of the optimization procedure, then using the step size function we can find the value of  $A$ ,  $A_{max} = \alpha_0$ , and using Eq. (23), we obtain the flowing expression

$$A_{max} = \frac{\delta}{E \|T_\mu \tilde{g}_0\| + 2\sqrt{\text{Var} \|T_\mu \tilde{g}_0\|}} \tag{24}$$

In which the expectation and variance of  $\|J_i \tilde{g}_0\|$  are calculated using the following formulas:

$$E \|T_\mu \tilde{g}_0\| = \frac{1}{N} \sum_{p_i \in I_i} \|T_\mu \tilde{g}_0\| \tag{25}$$

$$\text{Var} \|T_\mu \tilde{g}_0\| = \frac{1}{N-1} \sum_{p_i \in I_i} \left( \|T_\mu \tilde{g}_0\| - E \|T_\mu \tilde{g}_0\| \right)^2 \tag{26}$$

The choice of  $\lambda$  must be in a way to have a function decay that decreases neither drastically nor slowly, for this reason we let vary the parameter  $\lambda$  at each iteration in an interval that corresponds to a reasonable decay.

Consider  $S_k = \{\lambda_k^l, \lambda_k^c, \lambda_k^r\}$  is a varying decay parameter set at iteration  $k$ ,  $\lambda_k^l$  is the left decay parameter of the decay set in the sense of drastic decay,  $\lambda_k^r$  is the right decay parameter

in the sense of the slow decay and  $\lambda_k^c$  is the center of the decay parameter set. The decay parameters set are adjusted automatically at each iteration; the self-adjustment is performed according to the following adaptive logic steps:

i. Adjusting the center decay parameter

$$\lambda_{k+1}^c = w_k^l \lambda_k^l + w_k^r \lambda_k^r \tag{27}$$

Where  $\{w_k^l, w_k^r\}$  are respectively the model probabilities of the left and right decay parameters calculated as follow:

$$w_k^r = \frac{L(T(p, \mu_{\alpha_k^l}); \pi, m, \Sigma)}{L(T(p, \mu_{\alpha_k^r}); \pi, m, \Sigma) + L(T(p, \mu_{\alpha_k^l}); \pi, m, \Sigma)} \tag{28}$$

$$w_k^l = \frac{L(T(p, \mu_{\alpha_k^r}); \pi, m, \Sigma)}{L(T(p, \mu_{\alpha_k^r}); \pi, m, \Sigma) + L(T(p, \mu_{\alpha_k^l}); \pi, m, \Sigma)} \tag{29}$$

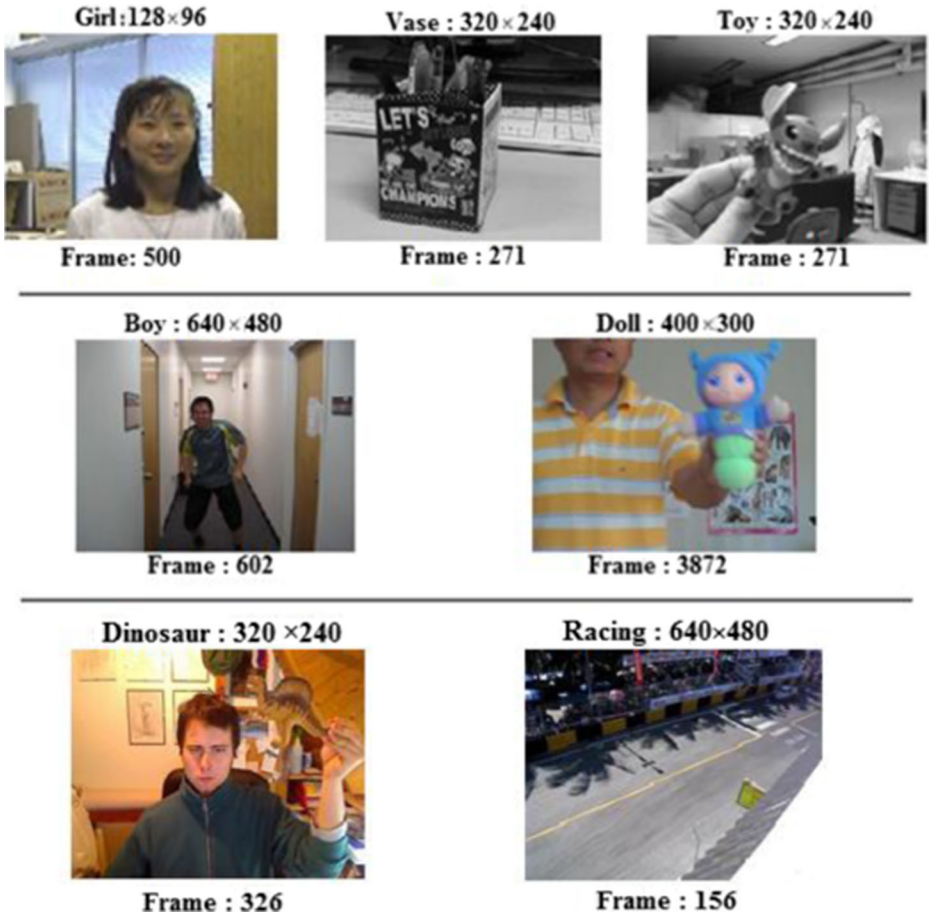


Fig. 6 Test sequences used in current evaluation

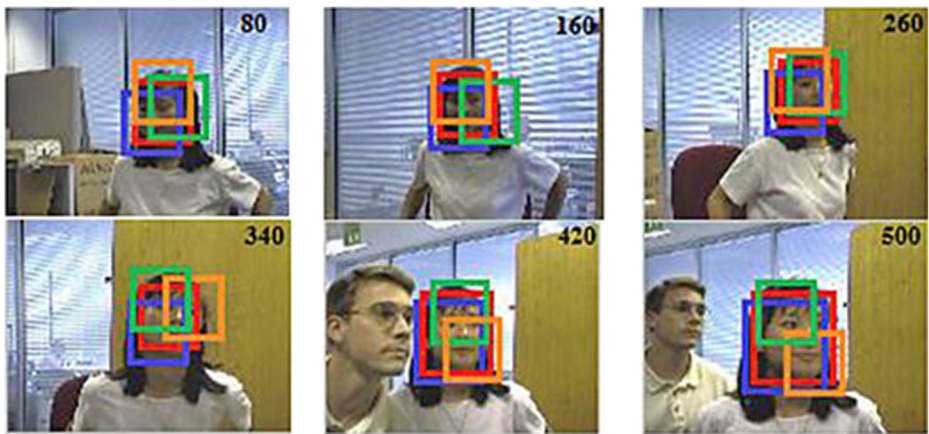


Fig. 7 Tracking results of the Girl sequence

Where  $\mu_{\alpha_k^r}$  is the right parametric motion model update associated with the right step size,  $\mu_{\alpha_k^l}$  is the left parametric motion model update associated with the left step size.

ii. Adjusting the left and right decay parameters

Case 1: when the decay rate does not show a significant change, the decay parameters set will keep stable, then

$$\lambda_{k+1}^l = \begin{cases} \lambda_{k+1}^c - \Delta\lambda_k^l / 2 & \text{if } w_k^l < t_1 \\ \lambda_{k+1}^c - \Delta\lambda_k^l & \text{else} \end{cases} \quad (30)$$

$$\lambda_{k+1}^r = \begin{cases} \lambda_{k+1}^c + \Delta\lambda_k^r / 2 & \text{if } w_k^r < t_1 \\ \lambda_{k+1}^c + \Delta\lambda_k^r & \text{else} \end{cases} \quad (31)$$

Where  $\Delta\lambda_k^l = \max(\lambda_k^c - \lambda_k^l, \Delta\lambda)$ ,  $\Delta\lambda_k^r = \max(\lambda_k^c - \lambda_k^r, \Delta\lambda)$ ,  $t_1 < 0.1$  is a threshold which allows to detect an invalid rate decay, and  $\Delta\lambda$  is the step jump of  $\lambda$ .

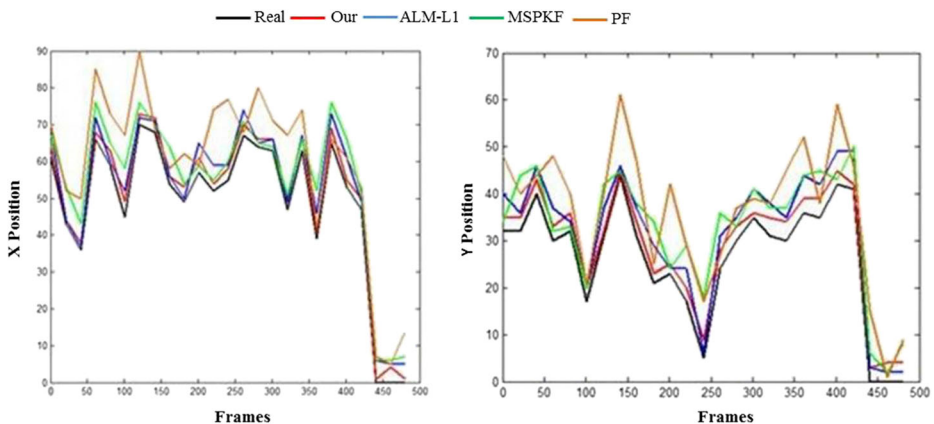


Fig. 8 Trajectory of the ground truth and the compared trackers in x, y direction of the Girl sequence

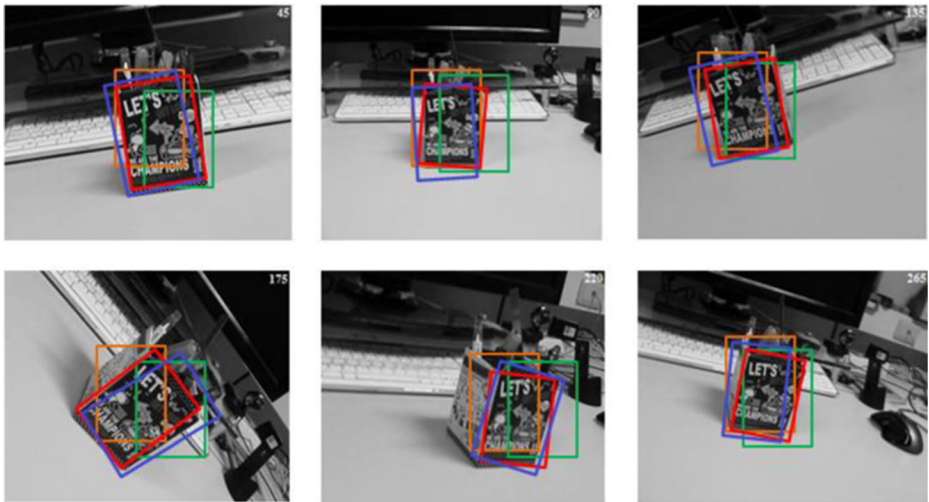


Fig. 9 Tracking results of the Vase sequence

Case 2: When the decay rate switches from left to right, so  $w_k^r$  is greater than  $w_k^l$ , then

$$\lambda_{k+1}^l = \lambda_{k+1}^c - \Delta\lambda_k^l \tag{32}$$

$$\lambda_{k+1}^r = \begin{cases} \lambda_{k+1}^c + 2\Delta\lambda_k^r & \text{if } w_k^r > t_2 \\ \lambda_{k+1}^c + \Delta\lambda_k^r & \text{else} \end{cases} \tag{33}$$

Where  $t_2 = 0.9$  is a threshold which allows detecting significant model.

Case 3: When the decay rate switch from right to left, so  $w_k^l$  is greater than  $w_k^r$ , then

$$\lambda_{k+1}^l = \begin{cases} \lambda_{k+1}^c - 2\Delta\lambda_k^l & \text{if } w_k^l > t_2 \\ \lambda_{k+1}^c - \Delta\lambda_k^l & \text{else} \end{cases} \tag{34}$$

$$\lambda_{k+1}^r = \lambda_{k+1}^c + \Delta\lambda_k^r \tag{35}$$

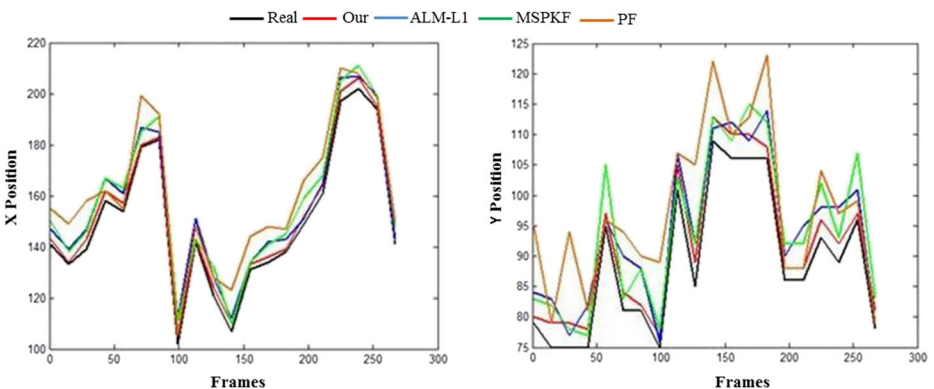


Fig. 10 Trajectory of the ground truth and the compared trackers in x, y direction of the Vase sequence

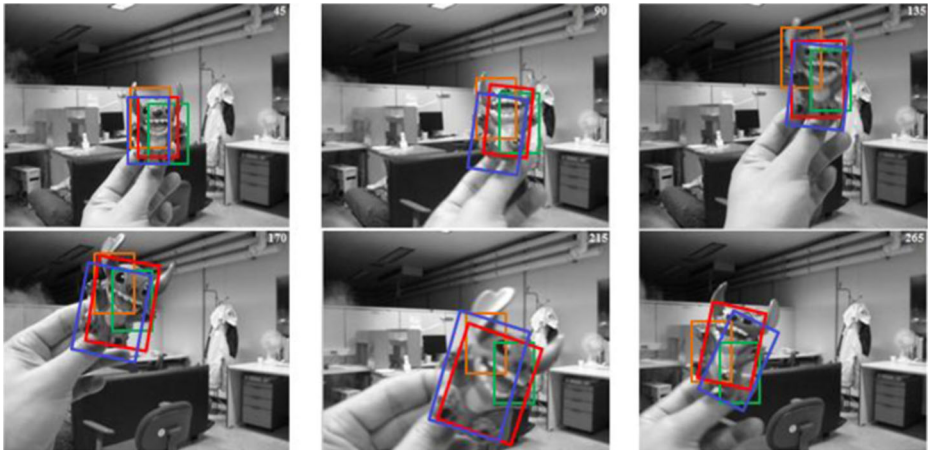


Fig. 11 Tracking results of the Toy sequence

## 5 Experiments

In this section, we conduct a series of qualitative and quantitative experiments to evaluate the performance and accuracy of our method using several public sequences<sup>1,2</sup> Fig. 6, that present different situations and challenging conditions. The length of sequences varies between 156 and 3872 frames and selected objects in each sequence were manually annotated by bounding boxes. All experiments were performed by computer equipped with a 2,10 Ghz Core 2 processor and a RAM of 2 Gb. Our method is compared with the most related methods with state of art tracking works, including particle filter with occlusion handling (PF) [18], visual object tracking based on mean shift and particle Kalman filter (MSPKF) [8] and the augmented Lagrange multiplier for robust visual tracking (ALM-L1) [22].

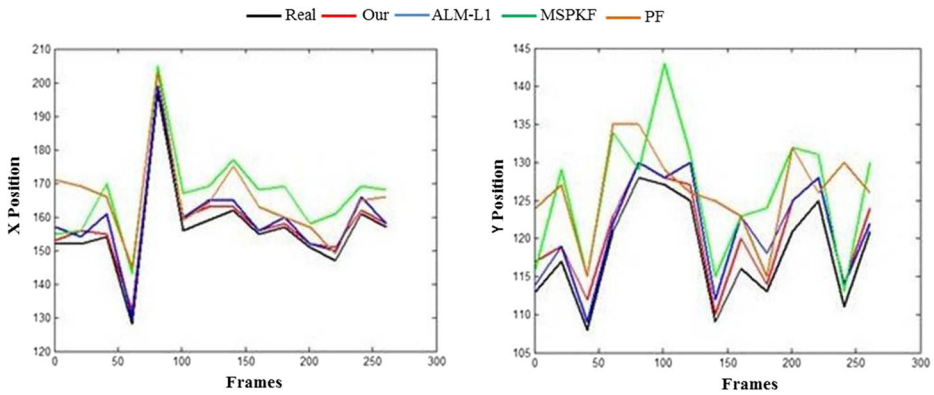
### 5.1 Qualitative results

In Fig. 7, Girl sequence, some significant frames of the tracking experiments results are exposed. The target moves forward and backward which causes a variation of scale and the appearance changes along the sequence, under this conditions all the trackers perform reasonable tracking despite the fact that the PF and the MSPKF can't estimate the scale change well, while the ALM-L1 and our method handle this challenge successfully. The comparison of different estimate trajectories with the ground truth one in Fig. 8 shows that the proposed tracker can keep a close and stable trajectory better than the methods of comparison.

In Fig. 9, Vase sequence, there are frequent variations of pose and scale, in addition the target rotates in the image plane, the proposed algorithm shows high performance, the ALM-L1 performs reasonably, the two methods can estimate the change in scale and orientation, although our tracker has excelled. The estimated trajectories are shown in Fig. 10, the proposed method performs the most accurate trajectory.

<sup>1</sup> [http://cvlab.hanyang.ac.kr/tracker\\_benchmark/datasets.html](http://cvlab.hanyang.ac.kr/tracker_benchmark/datasets.html)

<sup>2</sup> <http://votchallenge.net/vot2016/trackers.html>

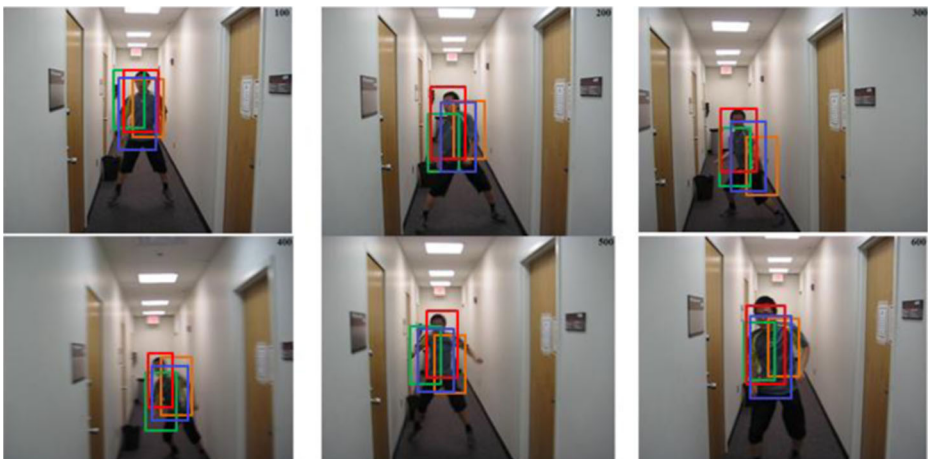


**Fig. 12** Trajectory of the ground truth and the compared trackers in x, y direction of the Toy sequence

In Fig. 11, Toy sequence, there is drastic scale and orientation change of the object, another challenge that makes tracking difficult is that the target moves fastly. The PF and MSPKF algorithms perform a fairly close tracking, despite they do not totally lose the target. While ALM-L1 and our method keep a close track and succeed to estimate the changes of scale and orientation. The comparison of the estimates trajectories and that of the ground truth in Fig. 12 shows the superiority and precision of our method.

In Fig. 13, Boy sequence. In addition to drastic changes of scale and orientation, the object moves suddenly and fastly, as well as the appearance of the target varies from one frame to another because of the blur motion. In these conditions, the PF and MSPKF algorithms drift away the truth ground center without loose the tracking. On the other hand, the ALM-L1 and our method perform reasonably well, with the superiority of the proposed algorithm. In Fig. 14, our tracker keeps a close track since it estimates the scale and the orientation change of the target well.

In the fifth sequence Doll, the tracking environment presents challenging conditions namely scale Variation, rotation and occlusion, some significant frames of the tracking results are exposed in Fig. 15, our method performs a reasonable tracking, even with scale and orientation changes, but when the target is occluded, the proposed tracker began to drift away from the ground truth area and lacks of precision, without any case losing the target, while ALM-L1



**Fig. 13** Tracking results of the Boy sequence

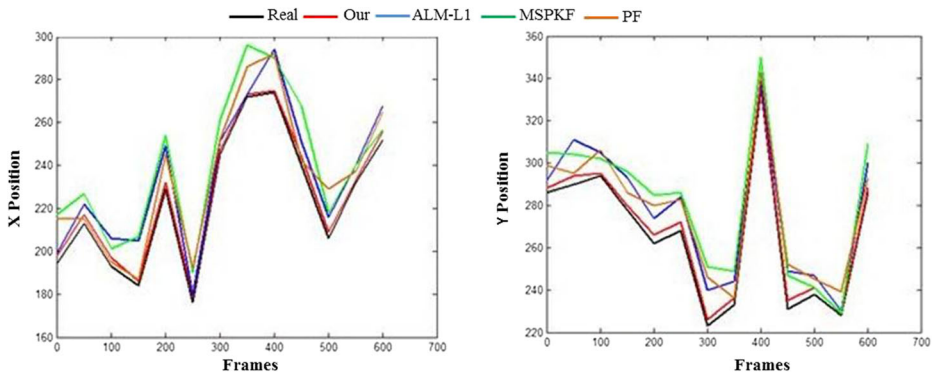


Fig. 14 Trajectory of the ground truth and the compared trackers in x, y direction of the Boy sequence

handles this challenge successfully, this is obvious since the proposed method does not manage the occlusions. Figure 16 presents the estimated trajectories. In the case where the object is not occluded the path estimated by our algorithm remains very close to the real one, on the other hand it cannot keep a stable tracking.

In the sixth sequence Dinosaur, the object moves in a challenging scene, characterized by drastic change of orientation, scale, brightness variation and background confusion. Figure 17 shows some significant frames. Both our method and ALM-L1 perform a precise tracking, and shows a good adaptation to the different environmental changes. The ground truth trajectory and those estimated by the different trackers are presented by the Fig. 18. The results obtained show that our method achieves a better correspondence than the methods of the comparison.

The last sequence Racing, there are frequent changes of orientation and scale, in addition the scene is affected by blur and brightness change. Despite the mentioned issues, the proposed tracker performs a better track than the comparison methods as shown in Fig. 19. In Fig. 20 the trajectory estimated by our method is well close to that of the ground truth. The proposed algorithm shows a better precision compared to the estimated trajectories of the methods of the comparison.

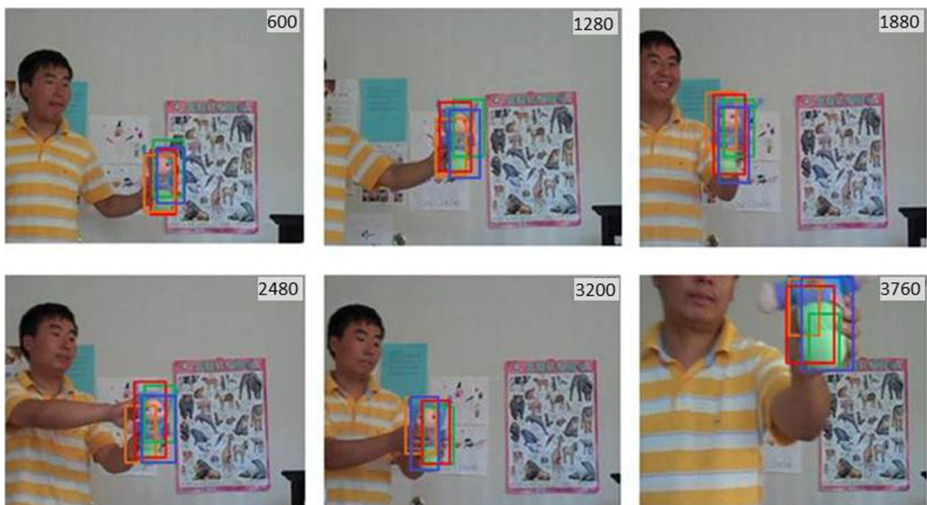


Fig. 15 Tracking results of the Doll sequence



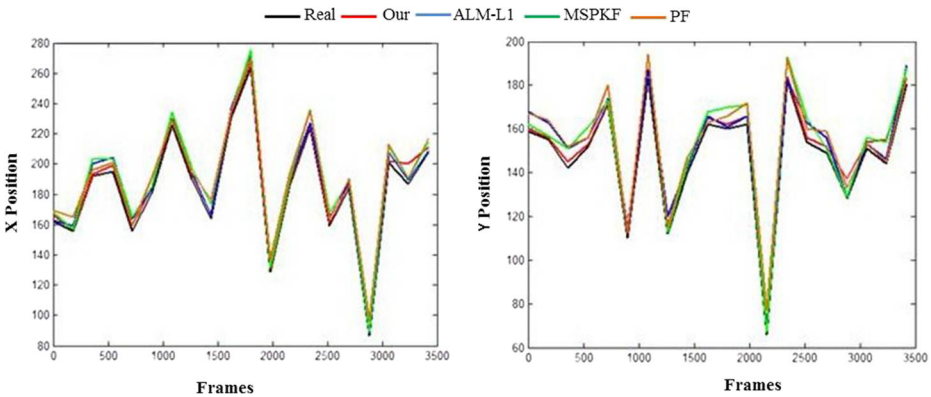


Fig. 16 Trajectory of the ground truth and the compared trackers in x, y direction of the Doll sequence

**5.2 Quantitative results**

To highlight the performance evaluation of our method, we consolidate our experience with quantitative evaluations. Different evaluation criteria were adopted, that were used in [10, 23]. The first criterion is the centroid position error, which is the Euclidian distance between the center of the estimate tracking result and that of the ground truth is calculated. Figures 21 and 22 show the plot of the center location error at each frame for the compared methods, in the majority of the sequences of the experiments, which present challenging conditions, like scale and orientation variations, the proposed tracker achieves the tracking with a low error and performs favorably against all the other methods. Table 1 summarizes the average center location errors in pixels Eq. (36). The numerical results confirm the superiority of our methods which achieves the smallest average errors; contrariwise, in the case of scene where the object is occluded, the error increases.

$$ANED = \frac{1}{N_{frames}} \sum_k \sqrt{\left(\frac{GA_x^k - T_x^k}{w}\right)^2 + \left(\frac{GA_y^k - TA_y^k}{h}\right)^2} \tag{36}$$

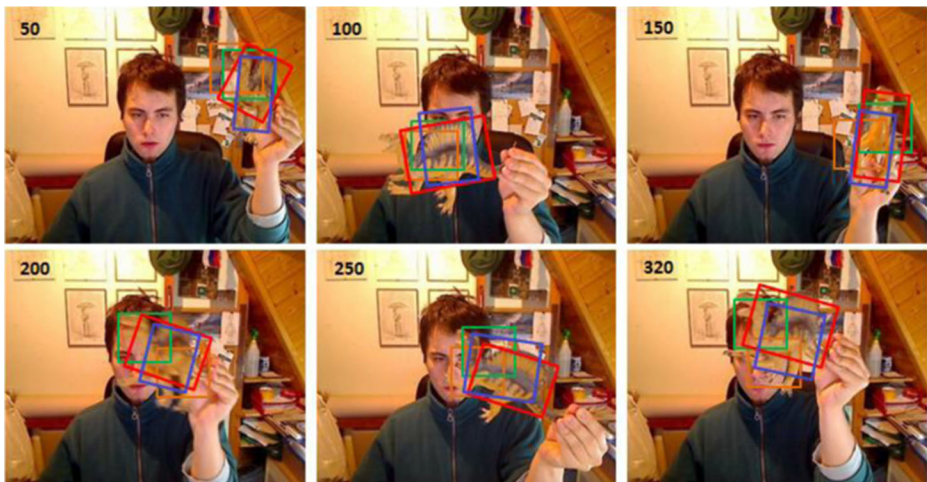


Fig. 17 Tracking results of the Dinosaur sequence

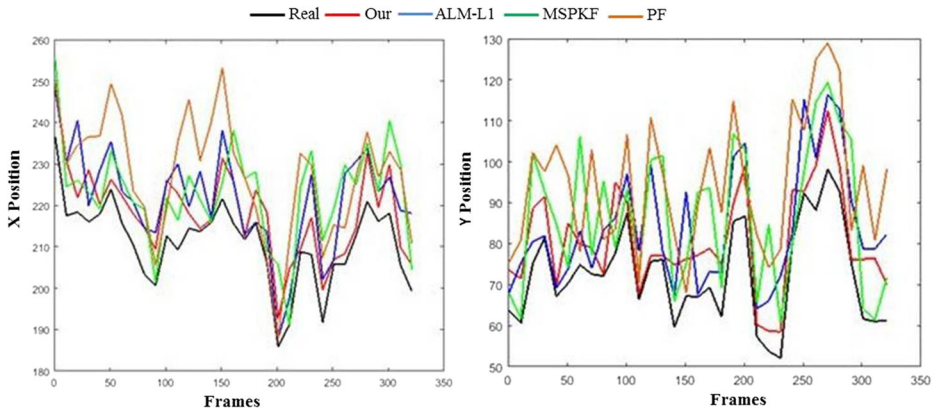


Fig. 18 Trajectory of the ground truth and the compared trackers in x, y direction of the Dinosaur sequence

Where  $TA_x^i, TA_y^i$  are respectively the x and the y position of the center of the object bounding box in frame k, and  $GA_x^i, GA_y^i$  are respectively the x and the y position of the center of the ground truth bounding box in frame k.

The next measures allow better evaluating of the performance of our algorithm. The quantitative results of the compared methods are presented in Table 2. The area-based F1-score defined by Eq. (37), this criteria allows to measure the accuracy of the tracking as much as the degree of matching between the estimated area and that of the ground truth, indeed it provides insight in the average coverage of the tracked bounding box and the ground truth bounding box.

$$F1 = \frac{1}{N_{frames}} \sum_k 2 \frac{|T^k \cap G^k|}{|T^k + G^k|} \tag{37}$$

Which  $T^k$  and  $G^k$  are defined by  $T^k = \frac{|TA^k \cap GA^k|}{|k|}$ ,  $G^k = \frac{|TA^k \cap GA^k|}{|GA^k|}$

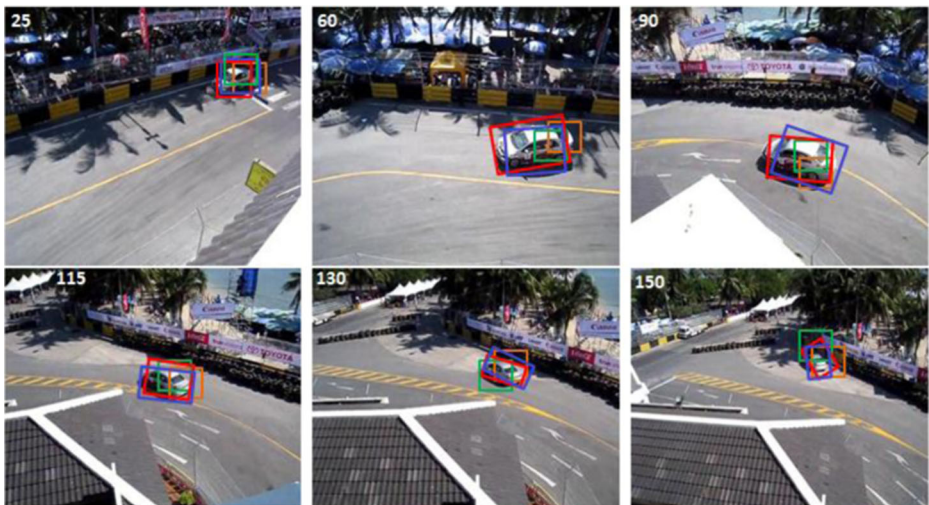
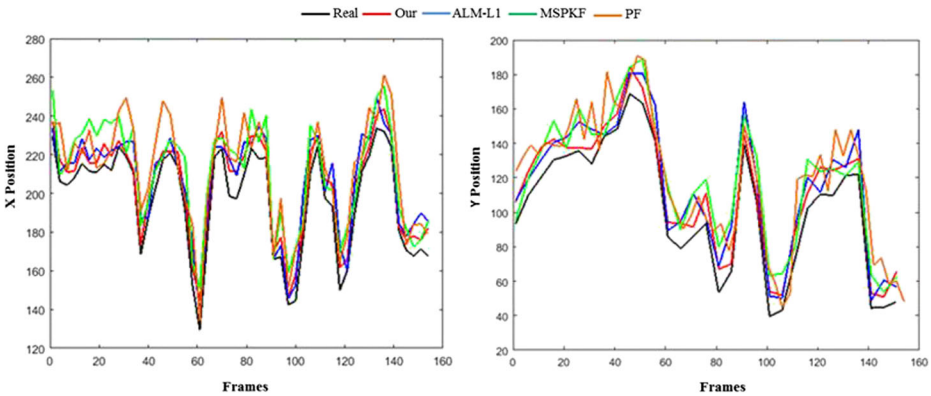


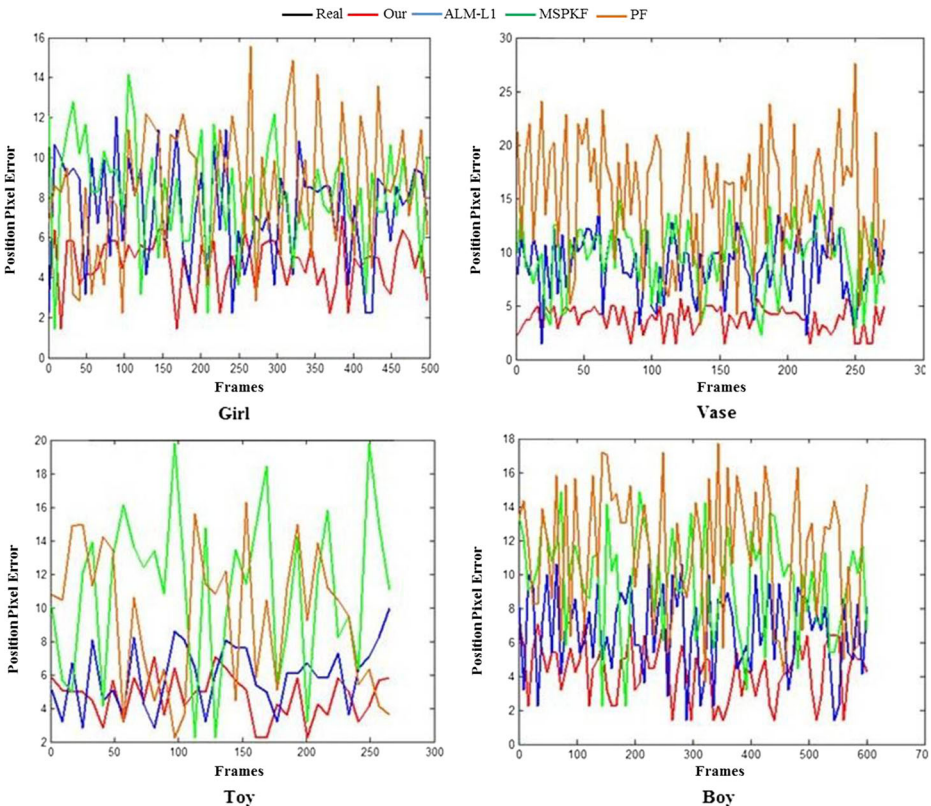
Fig. 19 Tracking results of the Racing sequence



**Fig. 20** Trajectory of the ground truth and the compared trackers in x, y direction of the Racing sequence

Where  $TA^k$  denotes the tracked object bounding box in frame  $k$ , and  $GA^k$  denotes the ground truth bounding box in frame  $k$ .

In the majority of the sequences our methods outperform the comparison methods, achieving the best precision in term of F1-score; indeed the average coverage of estimated rectangle exceeds 60% despite the challenging tracking conditions. In the case of



**Fig. 21** Performance of PF, MSPKF, ALM-L1 and the proposed method in terms of position error

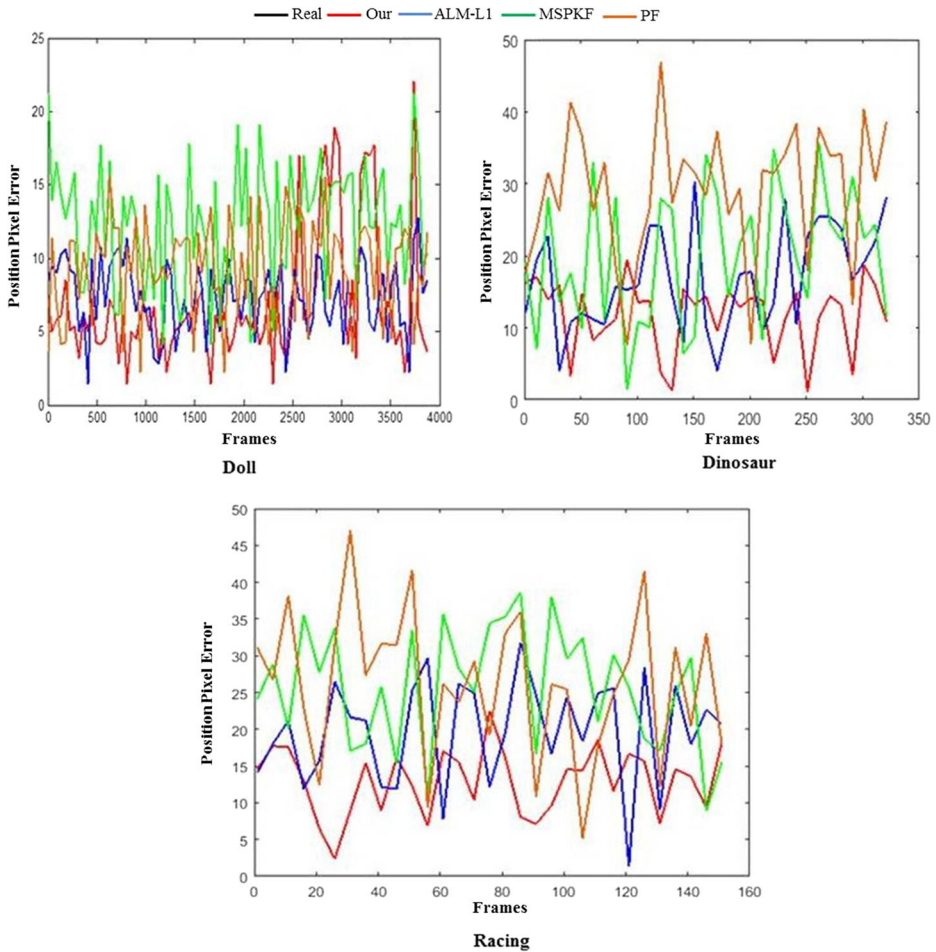


Fig. 22 Performance of PF, MSPKF, ALM-L1 and the proposed method in terms of position error

Table 1 The average center location error in pixels

Sequence	N <sub>Frames</sub>	Challenging aspects	PF	MSPKF	ALM-L1	Our
Girl	500	Scale Variation, In-Plane Rotation	12.36	11.40	9.68	6.78
Vase	271	Scale Variation, Fast Motion, In-Plane Rotation, Camera motion	20.18	17.28	10.76	4.87
Toy	271	Scale Variation, Fast Motion, In-Plane Rotation	22.32	25.60	11.26	5.92
Boy	602	Scale Variation, Fast Motion, In-Plane Rotation, Motion Blur	15.51	12.80	9.50	5.57
Doll	3872	Scale Variation, In-Plane Rotation, Occlusion	19.12	29.19	14.01	16.54
Dinosaur	326	Scale Variation, In-Plane Rotation, Brightness variation	27.81	22.45	18.43	12.62
Racing	156	Scale Variation, In-Plane Rotation, Brightness variation, Blur	29.01	24.72	16.32	10.82

**Table 2** Tracking accuracy

Sequence	N <sub>Frames</sub>	Challenging aspects	PF	MSPKF	ALM-L1	Our
Girl	500	Scale Variation, In-Plane Rotation	51.49	56.42	61.37	76.92
Vase	271	Scale Variation, Fast Motion, In-Plane Rotation, Camera motion	38.22	48.77	64.52	72.57
Toy	271	Scale Variation, Fast Motion, In-Plane Rotation	50.41	45.30	74.15	81.78
Boy	602	Scale Variation, Fast Motion, In-Plane Rotation, Motion Blur	47.54	54.46	65.07	74.17
Doll	3872	Scale Variation, In-Plane Rotation, Occlusion	52.17	38.72	62.48	59.36
Dinosaur	326	Scale Variation, In-Plane Rotation, Brightness variation	34.19	47.85	51.37	62.71
Racing	156	Scale Variation, In-Plane Rotation, Brightness variation, Blur	32.68	46.71	58.92	69.14

The area-based F1-score(%) is presented for the compared methods

scene when the tracked object is occluded, the coverage of the proposed tracker is disturbed.

To measure the accuracy of the tracking, the ATA metric is used, the quantitative results of the compared methods are presented in Table 3. The ATA accuracy metric defined by Eq. (38) indicates how much tracking bounding boxes overlap with the ground truth bounding boxes. An object is considered to be correctly tracked in a frame if the estimated rectangle covers at least 25% of the area of the target in the ground truth.

$$ATA = \frac{1}{N_{frames}} \sum_i \frac{|O^i \cup GO^i|}{|O^i \cap GO^i|} \quad (38)$$

Where  $O^i$  denotes the tracked object bounding box in frame  $i$ , and  $GO^i$  denotes the ground truth bounding box in frame  $i$ .

Overall, our tracker produces the best measures in terms of ATA, indeed the numerical results are lower than 1.9, therefore the estimated bounding boxes overlap well with the ground truth one, thus

**Table 3** Tracking accuracy

Sequence	N <sub>Frames</sub>	Challenging aspects	PF	MSPKF	ALM-L1	Our
Girl	500	Scale Variation, In-Plane Rotation	22,481	21,352	15,946	118,296
Vase	271	Scale Variation, Fast Motion, In-Plane Rotation, Camera motion	31,421	28,323	19,815	16,190
Toy	271	Scale Variation, Fast Motion, In-Plane Rotation	26,053	29,856	17,982	1,0231
Boy	602	Scale Variation, Fast Motion, In-Plane Rotation, Motion Blur	30,195	28,874	20,904	15,070
Doll	3872	Scale Variation, In-Plane Rotation, Occlusion	30,024	32,455	19,793	20,349
Dinosaur	326	Scale Variation, In-Plane Rotation, Brightness variation	40,137	31,843	31,268	18,349
Racing	156	Scale Variation, In-Plane Rotation, Brightness variation, Blur	42,188	35,287	29,167	17,731

The average tracking accuracy (ATA) between the estimated tracked object bounding box and the ground truth bounding box is presented for the compared methods

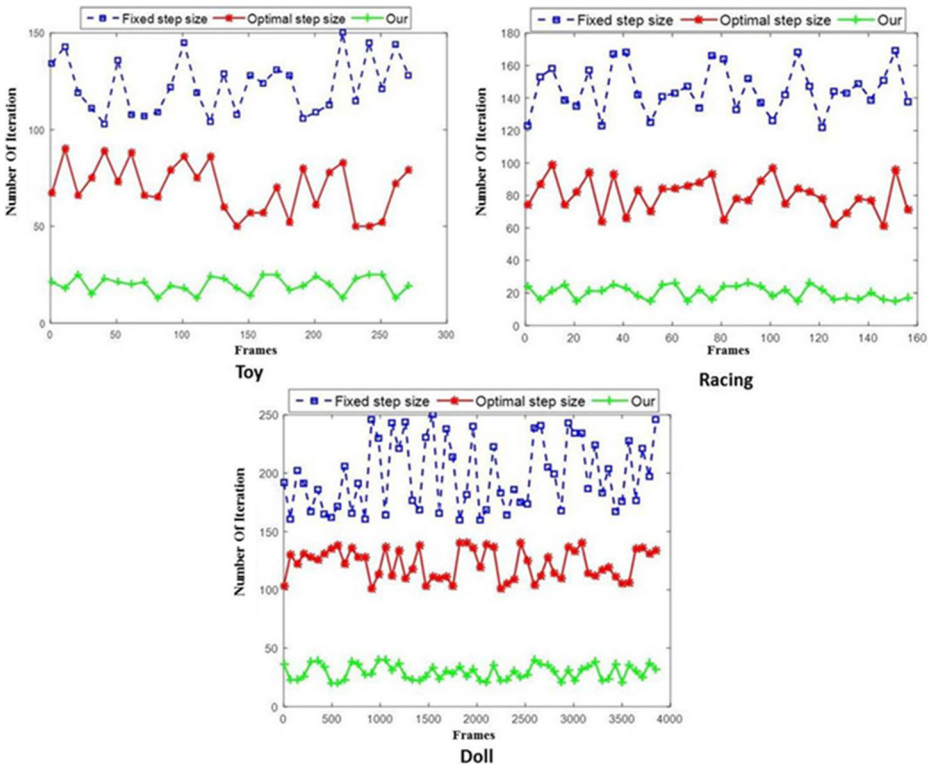
**Table 4** Time efficiency

Sequence	$N_{Frames}$	PF	MSPKF	ALM-L1	Our
Girl	500	0.0546	0.0831	0.0641	0.0420
Vase	271	0.1037	0.1738	0.1322	0.0805
Toy	271	0.0804	0.1205	0.0994	0.0669
Boy	602	0.0926	0.1506	0.1147	0.0758
Doll	3872	0.9905	0.1692	0.1270	0.0791
Dinosaur	326	0.0917	0.1480	0.1094	0.0701
Racing	156	0.1503	0.1927	0.1805	0.0836

Average execution times for the compared methods (s/frame)

performing a better accuracy than the other trackers. However, in the case of Doll sequence which presents occlusion, our method performs weakly. For example between the frames 22,632 2239 and between 2552 and 2573, the target is occluded by a hand, thus the tracking error increase, hence the overlap area between the estimated bounding boxe and the ground truth one starts to decrease. Then our method achieves a tracking accuracy result less than ALM-L1 method.

We close this part of experimentation by presenting the average running time, the numerical results are exposed in the Table 4, our method shows a better average computation time than the comparison method, in fact the use of the descent gradient



**Fig. 23** Comparing the number of iterations of different step size gradient descent methods

as an optimization method as well as the automatic step size adjustment strategy, makes the method fast without losing anything in convergence accuracy. In addition the modeling of the appearance and update is achieved with a low complexity, as a result our method achieves a reasonable computation time that promotes its implementation in real-time applications.

### 5.3 Convergence

The results of the experiments performed on different challenging sequences; show that the proposed method achieves a precise and stable tracking in a reduced running time. These performances are well justified; indeed the method proposed is based on the optimization gradient descent with adaptive step size. The strategy adopted to update the displacement step allows the algorithm achieving a convergence to an optimal location in a fewer iterations, therefore achieving a very low location error.

The formulation of the cost function is based on a summation of a continuous and convex usual functions; this implies the convergence of the algorithm into a finite number of iterations. We tested our tracking approach using three optimization methods based on the gradient descent, namely the gradient descent with fixed step, the gradient descent with optimal step, as the proposed adaptive decaying step size displacement. Figure 23 shows the results of comparisons in terms of number of iteration for the sequences Toy, Racing and Doll. The adaptive decaying strategy, achieves a very small number of iterations compared to other methods used in this comparison, hence allowing a fast convergence.

## 6 Conclusion

In summary, we have present a new real-time tracking method, the appearance is modeled using the GMM and the motion of object is modeled in the image plane by a planar parametric linear motion model consisting of a translation and a rotation, and scaling. Then the position detection is ensured by an optimization strategy based on the gradient descent method with an adaptive step size scheme, indeed the step size is adjusted automatically at each iteration to ensure a fast resolution and accurate convergence. The experiments conducted on various challenging sequences, which present different challenging conditions, validated the efficiency and better tracking accuracy of the proposed tracker especially in case of scale variation and when the target rotates in the image plane. In addition the numerical results in term of average running times show the computational time efficiency, which encourage the use of the proposed algorithm in real time applications.

## References

1. Ahmad A, Abdul J, Jianwei N, Xiaoke Z, Saima R, Javed A, Muhammad AI (2016) Visual object tracking—classical and contemporary approaches. *Frontiers of Computer Science* 10(1):167–188
2. Avidan S (2007) Ensemble tracking. *IEEE Trans Pattern Anal Mach Intell* 29(2):261–271
3. Choe G, Wang T, Liu F, Choe C, Jong M (2015) An advanced association of particle filtering and kernel based object tracking. *Multimed Tools Appl* 74(18):7595–7619
4. Coifman B, Beymer D, McLauchlan P (1998) A real-time computer vision system for vehicle tracking and traffic surveillance. *Transportation Research Part C: Emerging Technologies* 6(4):271–288

5. Collins RT, Liu Y, Leordeanu M (2005) Online selection of discriminative tracking features. *IEEE Trans Pattern Anal Mach Intell* 27(10):1631–1643
6. Cui J, Liu Y, Xu Y, Zhao H, Zha H (2013) Tracking generic human motion via fusion of low- and high-dimensional approaches. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 43(4):996–1002
7. Dhassi Y, Aarab A (2018) Visual tracking based on adaptive interacting multiple model particle filter by fusing multiples cues. *Multimed Tools Appl*:1–34
8. Iswanto IA, Li B (2017) Visual object tracking based on mean-shift and particle-Kalman filter. *Procedia Computer Science* 116:587–595
9. Jeong J, Yoon TS, Park JB (2017) Mean shift tracker combined with online learning-based detector and Kalman filtering for real-time tracking. *Expert Syst Appl* 79:194–206
10. Karasulu B, Korukoglu S (2011) A software for performance evaluation and comparison of people detection and tracking methods in video processing. *Multimed Tools Appl* 55(3):677–723
11. Karavasilis V, Nikou C, Likas A (2015) Visual tracking using spatially weighted likelihood of Gaussian mixtures. *Comput Vis Image Underst* 000:1–15
12. Klein S, Plum J, Staring M, Viergever M (2009) Adaptive stochastic gradient descent optimisation for image registration. *Int J Comput Vis* 81(3):227–239
13. Kong J, Liu C, Jiang M, Wu J, Tian S, Lai H (2016) Generalized tP-regularized representation for visual tracking. *Neurocomputing* 213:155–161
14. Leichter I, Lindenbaum M, Rivlin E (2010) Mean shift tracking with multiple reference color histograms. *Comput Vis Image Underst* 114:400–408
15. Li G, Liang D, Huang Q, Jiang S, Gao W (2008) Object tracking using incremental 2D-LDA learning and Bayes inference. In: *Image Processing. ICIP 2008. 15th IEEE International Conference on*
16. Li X, Hu W, Shen C et al (2013) A survey of appearance models in visual object tracking. *ACM Transactions on Intelligent Systems and Technology* 4(4):58
17. Li P, Wang D, Wang L, Lu H (2018) Deep visual tracking: review and experimental comparison. *Pattern Recogn* 76:323–338
18. Lin SD, Lin J-J, Chuang C-Y (2015) Particle filter with occlusion handling for visual tracking. *IET Image Process* 9(11):959–968
19. Liu Y, Cui J, Zhao H, Zha H (2012) Fusion of low-and high-dimensional approaches by trackers sampling for generic human motion tracking. In: *21st international conference on pattern recognition (ICPR 2012)*, Tsukuba, Japan
20. Liu Z, Song Y-q, Xie C-h, Tang Z (2016) A new clustering method of gene expression data based on multivariate Gaussian mixture models. *SIViP* 10(2):359–368
21. Pan Z, Liu S, Sangaiah AK, Muhammad K (2018) Visual attention feature (VAF) : a novel strategy for visual tracking based on cloud platform in intelligent surveillance systems. *Journal of Parallel and Distributed Computing* 120:182–194
22. Shi Y, Zhao Y, Deng N, Yang K (2015) The augmented Lagrange multiplier for robust visual tracking with sparse representation. *Optik* 126:937–941
23. Smeulders AWM, Chu DM, Cucchiara R (2014) Visual tracking: an experimental survey. *IEEE Trans Pattern Anal Mach Intell* 36(7):1442–1468
24. van Mourik MJW, Zaar DVJ, Smulders MW, Heijman J, Lumens J, Dokter JE, Passos VL, Schalla S, Knackstedt C, Schummers G, Gjesdal O, Edvardsen T, Bekkers SCAM (2018) Adding speckle-tracking echocardiography to visual assessment of Systolic Wall motion abnormalities improves the detection of myocardial infarction. *J Am Soc Echocardiogr* 32:65–73
25. Villagra J, Acosta L, Artuñedo A, Blanco R, Clavijo M, Fernández C, Godoy J, Haber R, Jiménez F, Martínez C, Naranjo JE, Navarro PJ, Paúl A, Sánchez F (2018) Automated driving. In: *Intelligent vehicles, enabling technologies and future developments*, pp 275–342
26. Vysochanskij DF, Petunin YI (1980) Justification of the  $3\sigma$  rule for unimodal distributions. *Theory of Probability and Mathematical Statistics* 21:25–36
27. Wang H (2015) Adaptive visual tracking for robotic systems without image-space velocity measurement. *Automatica* 55:294–301
28. Yang W, Zhao M, Huang Y, Zheng Y (2018) Adaptive online learning based robust visual tracking. *IEEE Access* 6:14790–14798
29. Ye L, Cui J, Zhao H, Zha H (2012) Fusion of low-and high-dimensional approaches by trackers sampling for generic human motion tracking. In: *Proceedings of the 21st international conference on pattern recognition (ICPR2012)*
30. Yu W, Hou Z, Hu D, Wang P (2017) Robust mean shift tracking based on refined appearance model and online update. *Multimed Tools Appl* 76(8):10973–10990
31. Zhi-Qiang H, Xiang L, Wang Sheng Y, Wu L, An Qi H (2014) Mean-shift tracking algorithm with improved background-weighted histogram. In: *Intelligent systems design and engineering applications (ISDEA)*



32. Zhou Z, Zhou M, Li J (2017) Object tracking method based on hybrid particle filter and sparse representation. *Multimed Tools Appl* 76(2):2979–2993

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Younes Dhassi** Received his advanced graduate degree (DESA) in 2008 from faculty of sciences Dhar- El mahraz (FSDM) Fes Morocco. He received his Phd degree in 2018 from the University Sidi Mohamed Ben Abdellah, His research areas include, computer learning and visual tracking.



**Abdellah Aarab** is Professor at the Faculty of Sciences, Fes Morocco. He received his “Doctorat de 3ième cycle” degree from the University of Pau, France in 1982 and “Thèse d’Etat” degree in 1999 from university Sidi Mohamed Ben Abdellah Fes Morocco. His research areas include, image processing, information retrieval and industrial applications.