Object Tracking based on Relaxed Inverse Sparse Representation

Junxing Zhang[1], Chunjuan Bo[2]*, Jianbo Tang[2], Peng Song[2]
1. College of Information & Communication Engineering, Dalian Nationalities University, Dalian 116600, China
[e-mail: zhangjunxing@dlnu.edu.cn]
2. College of Electromechanical Engineering, Dalian Nationalities University, Dalian 116600, China
[e-mail: bcj@dlnu.edu.cn, tjb@dlnu.edu.cn, spony@dlnu.edu.cn]
*Corresponding author: Chunjuan Bo

Received January 12, 2015; revised May 17, 2015; accepted July 12, 2015;
published September 30, 2015

Abstract

In this paper, we develop a novel object tracking method based on sparse representation. First, we propose a relaxed sparse representation model, based on which the tracking problem is casted as an inverse sparse representation process. In this process, the target template is able to be sparsely approximated by all candidate samples. Second, we present an objective function that combines the sparse representation process of different fragments, the relaxed representation scheme and a weight reference prior. Based on some propositions, the proposed objective function can be solved by using an iteration algorithm. In addition, we design a tracking framework based on the proposed representation model and a simple online update manner. Finally, numerous experiments are conducted on some challenging sequences to compare our tracking method with some state-of-the-art ones. Both qualitative and quantitative results demonstrate that the proposed tracking method performs better than other competing algorithms.

Keywords: Object tracking; robust tracking; sparse representation; relaxed representation

This work is supported by Natural Science Foundation of Liaoning Province (Grant No. 2013020018), General Science Research Projects of Liaoning Provincial Department of Education (Grant No. L2014536), Fundamental Research Funds for Central Universities (Grant No. DC201501010401，DC201501060201).

http://dx.doi.org/10.3837/tiis.2015.09.020 **ISSN : 1976-7277**

1. Introduction

As one of the important problems in computer vision and pattern recognition, object tracking plays a critical role in many research lines (e.g., motion analysis, video compression and activity recognition) and has many useful applications in realistic scene (e.g., traffic control, human computer interface and video surveillance) [1][2]. The traditional tracking methods often work well under some well-controlled conditions or track some specific objects [3][4] (such as human, car, face and so on); while online visual tracking aims to track any object in realistic conditions [5]. It is very difficult to develop an effective online tracking method for many challenging factors [6], which mainly include illumination variation, pose change, partial occlusion, scale change, background clutter and so on.

From the perspective of adopted theories and techniques, online visual tracking algorithms can be categorized into three classes: tracking methods based on state estimation, tracking methods based on online classifiers and tracking methods based on template matching. First, tracking methods based on state estimation consider the tracking problem as a state estimation problem and reclusively estimate the states of the tracked target, such as Kalman filter [7], particle filter [8][9] and so on. This type of tracking methods mainly focus on designing an effective motion model and lacks of the discussion of a robust appearance model, thus, leads to an unstable tracking performance. Second, tracking methods based on online classifiers (usually called discriminative trackers) treat the tracking problem as a local detection problem, which aims to distinguish the tracked object from its local surroundings and learn robust online classifiers to capture appearance changes of both object and background during the tracking process. Thus, many classical and state-of-the-art machine learning algorithms can be used to slove the tracking problem, including support vector tracking (SVT) [10], ensemble tracking (EST) [11], online boosting tracking (OBT) [12][13], semi-supervised boosting tracking (SemiBT) [14], multiple instance learning (MIL) [15], tracking-learning-detection (TLD) [16], to name a few. However, this kind of trackers usually achieves not good performance in terms of accuracy since the number of collected positive and negative samples is limited in the tracking process.

During the tracking process, tracking methods based on template matching search for a most likely image region being of the highest similarity or the smallest distance to the tracked object. In 1981, Lucas and Kanade [17] propose an iterative image registration method, which is the basis of the optical flow tracking algorithm. In 2003, Comaniciu *et al.* [7] present a kernel-based tracking framework, which exploits a spatial kernel function to measure the similarities between the tracked object and candidates and uses the Mean Shift method to achieve a fast matching. In 2008, Ross *et al.* [18] propose an incremental visual tracking (IVT) method based on online subspace learning, which learns an incremental principle component analysis (PCA) subspace in an online fashion. The IVT method is able to handle the illumination variation and pose change due to the PCA assumption, but is sensitive to outliers (such as partial occlusion and background clutter). Kwon and Lee [19] adopt a sparse PCA method to select multiple color and edge templates, which is robust to many challenging cases (such as illumination variation, scale change, pose change, non-rigid motion and so on). However, this method is too complex to be applied in real tracking problems.

Recently, the sparse representation theory has been widely used in the fields of image processing and computer vision [20]. Motivated by the success of sparse representation for face recognition [21], Mei *et al.* [22] introduce sparse representation into the tracking filed and propose a L1 tracker, which uses a series of object and trivial templates to sparsely represent the tracked object. After that, many researchers improve the original L1 tracker in terms of both speed and accuracy. Based on the original L1 tracker, Mei *et al.* [23] compute a minimum error bound of each candidate by using the L2-norm minimization problem and discard those candidates with large reconstruction errors, which effectively reduces the number of complicated L1-norm minimizations in each frame. Then, Bao *et al.* [24] introduce an accelerate proximal gradient (APG) method to speed up the solution process of the L1-norm minimization. Besides, a lot of researchers have also attempted to improve the L1 tracker in different aspects, such as considering both positive and negative templates [25], adopting different optimization techniques [26], modeling the relationships among different candidates [27], combining subspace and sparse representation models [28][29] and so on.

Inspired by the "template matching"-based trackers (especially the "sparse representation"-based ones), this paper presents a novel tracking method based on the proposed relaxed inverse sparse representation model. The contributions of this work are mainly four folds. First, we treat the tracking problem as an inverse sparse representation process and propose a novel objective function to depict this idea. The proposed objective function integrates the sparse representation, relaxed representation and weight prior in a unified framework. Second, we design an iteration algorithm to effectively solve our objective functions based on three propositions. In addition, the proposed representation model is embed into a Bayesian inference framework for designing a robust tracker, in which a simple template update method is introduced. Finally, many experiments are conducted on some challenging image sequences to compare the proposed tracking method with other state-of-the-art trackers. The experimental results demonstrate that our tracker achieves good performance than other tracking methods.

The rest of this paper is organized as follows. Section 2 introduces the proposed tracking framework with the inverse sparse representation method, including motivation, problem formulation, objective function and so on. In Section 4, some experiments are conducted to evaluate the proposed tracker and compare it with many state-of-the-art algorithms. Finally, Section 5 concludes this paper.

2. Object Tracking based on Relaxed Inverse Sparse Representation

2.1 Motivation

This paper is motivated by the recent success of sparse representation in visual tracking [19] and object recognition [27] [28]. The basic idea of the original L1 tracker [19] is that each candidate $\mathbf{d}$ can be sparsely represented by a set of object $(\mathbf{T})$ and trivial templates $(\mathbf{I})$, in which the coding coefficient vector can be solved by the following L1 minimization problem, i.e.,

$$\min_{\mathbf{x}} \left\| \mathbf{d} - [\mathbf{T},\mathbf{I}]\mathbf{c} \right\|_2^2 + \lambda \left\| \mathbf{c} \right\|_1 \tag{1}$$

in which $\mathbf{c}$ is the coding coefficient vector and $\|.\|_1$ encourage a sparse solution. However, in the tracking problem, it requires to maintain many candidates to approximate the probability of the object's state. Thus, it needs to calculate many L1 minimization problems, which will make the tracker very slow. In recent, some researchers have studied another line in sparse representation, sparsity induced similarity [27]. This research line treats the solution obtained by sparse representation as a similarity measurement. From this view, the tracking problem can be viewed as an inverse sparse representation process, i.e., representing the template by using a set of candidates rather than coding each candidate by using templates. The core idea of the inverse sparse representation process is illustrated in **Fig. 1**.

$$\mathbf{y} \approx \mathbf{D}\mathbf{x} \tag{2}$$

We note that the coefficient vetcor $\mathbf{x}$ can be viewed as the similarity degrees between different candidates and the object template according to the sparsity induced similarity framework[27]. Compared with the original L1 tracker [19], this idea merely sloves one L1 minination problem (i.e., $\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{D}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_1, \mathbf{x} \geq \mathbf{0}$) to determine the likelihood values of different candidates. In the tracking process, the target's appearance may experience some unexpected noises or outliers, such as partial occlusion, local illumination variation and so on. However, the holistic representation in equation (2) cannot deal with this dilemma. Thus, to alleviate this problem, we divide the observation patch (for both object template $\mathbf{y}$ and candidates $\mathbf{D}$ into M fragments), and then the representation process in equation (2) can be converted into M sub-processes $\mathbf{y}_m \approx \mathbf{D}_m\mathbf{x}_m$. But in this fragment scheme, different fragments are treated to be of equal importance and therefore cannot make the tracker aviod the effect of outliers. To emphasize the differences of different fragments and avoids the effect of outliers (such as partial occlusion), the representation coefficient vectors $\mathbf{x}_m|_{m=1}^{M}$ should be similar but not same [28]. This idea can be described by using the term $\sum_{m=1}^{M} w_m \|\mathbf{x}_m - \bar{\mathbf{x}}\|_2^2$, which $\bar{\mathbf{x}}$ is a weighted average vector to make different coefficient vectors be similar and $w_m|_{m=1}^{M}$ are weights of different fragments for depicting the differences among coefficient vectors. Based on the above-mentioned discussions, we present the objective function of the proposed representation model in the next subsection. Based on the proposed model, i.e., the relaxed inverse sparse representation, the overall framework of the tracking method is illustrated in **Fig. 1.**

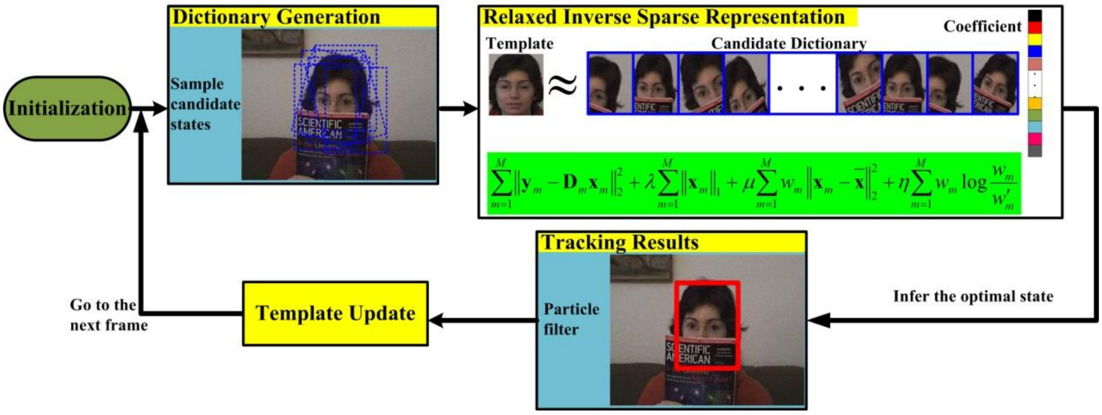

Initialization
Dictionary Generation
Sample candidate states
Relaxed Inverse Sparse Representation
Template
Candidate Dictionary
Coefficient
≈
$\sum_{m=1}^{M}\|\mathbf{y}_m-\mathbf{D}_m\mathbf{x}_m\|_2^2+\lambda\sum_{m=1}^{M}\|\mathbf{x}_m\|_1+\mu\sum_{m=1}^{M}w_m\|\mathbf{x}_m-\bar{\mathbf{x}}\|_2^2+\eta\sum_{m=1}^{M}w_m\log\frac{w_m}{w'_m}$
Tracking Results
Particle filter
Infer the optimal state
Template Update
Go to the next frame

Fig. 1. The overall framework of the proposed tracking algorithm.

2.2 Problem formulation

In this work, we cast the tracking problem as a relaxed inverse sparse representation problem, the core objective function of which can be defined as

$$J\left(\bar{\mathbf{x}},\mathbf{x}_m\Big|_{i=1}^{M},\mathbf{w}\right)=\sum_{m=1}^{M}\left\|\mathbf{y}_m-\mathbf{D}_m\mathbf{x}_m\right\|_2^2+\lambda\sum_{m=1}^{M}\left\|\mathbf{x}_m\right\|_1+\mu\sum_{m=1}^{M}w_m\left\|\mathbf{x}_m-\bar{\mathbf{x}}\right\|_2^2+\eta\sum_{m=1}^{M}w_m\log\frac{w_m}{w'_m} \quad (3)$$

where the first term denotes the reconstruction error based on coding coefficient vectors, the second term is the L1 regularization term that aims to encourage the coding coefficients being sparse, the third term is the relaxed term that measures the consistency (or dis-consistency) among different coding coefficient vectors, and the last one is a Kullback–Leibler divergence term that makes sure the weight vector $\mathbf{w}$ should be more similar a reference weight vector $\mathbf{w}'$ (that is, $\mathbf{w}'$ provides a prior on $\mathbf{w}$), λ,μ,η are parameters to balance different terms.

We note that $\bar{\mathbf{x}}$ can be viewed as an overall sparisty induced similarity measure between the template $\mathbf{y}$ and the candidate dictionary $\mathbf{D}$. The optimal $\bar{\mathbf{x}}^*$ can be obtained by optimizing the following optimization problem,

$$\begin{aligned}&\min_{\bar{\mathbf{x}},\mathbf{x}_m|_{i=1}^{M},\mathbf{w}'} J\left(\bar{\mathbf{x}},\mathbf{x}_m\Big|_{i=1}^{M},\mathbf{w}\right)\\&s.t.\ \bar{\mathbf{x}}\geq\mathbf{0},\mathbf{x}_m\geq\mathbf{0},\mathbf{w}\geq\mathbf{0}\end{aligned} \quad (4)$$

We note that the objective function (3) is a unified framework to combine several key components, such as the inverse sparse representation process, the fragment scheme and the weight adaptive scheme. The inverse sparse representation process makes the tracker use different candidates to sparsely represent the object template, in which the representation coefficients can be viewed as the observation likelihood values of those candidates. The fragment scheme makes the inverse sparse representation process be in the fragment level rather than holistic level, which effectively exploit the differences of different fragments in representation process. In addition, the weight adaptive scheme could determine the important degrees of different fragments in an online manner. Thus, the combination of these key components in a unified objective function is able to benefit obtaining accurate observation likelihood during the tracking process. The effectiveness of these components can be demonstrated in the experiment section.

2.3 Problem solution

To the best of our knowledge, there is no closed-from solution for the optimization problem (4). So, we propose an iteration algorithm to estimate $\mathbf{x}_m\big|_{m=1}^{M}$, $\bar{\mathbf{x}}$ and $\mathbf{w}$ based on the following three propositions.

Proposition 1: Given the optimal solutions $\bar{\mathbf{x}}^*$ and $\mathbf{w}^*$, the optimal coefficient vectors $\mathbf{x}_m^*\big|_{m=1}^{M}$ can be obtained by solving M individual sparse coding problems, which can be solved by the LASSO method.

If the optimal values $\bar{\mathbf{x}}^*$ and $\mathbf{w}^*$ are given, the minimization of the problem (5) can be converted into the following optimization problem,

$$\begin{aligned} &\mathbf{x}_m^* |_{m=1}^M = \arg\min_{\mathbf{x}_m|_{m=1}^M} J^1\left(\mathbf{x}_m |_{m=1}^M\right) \\ &s.t.\, \mathbf{x}_m \ge 0,\, m = 1,2,\ldots,M \end{aligned} \tag{6}$$

where the objective function $J^1\left(\mathbf{x}_m |_{m=1}^M\right)$ is defined as

$$J^1\left(\mathbf{x}_m |_{m=1}^M\right) = \sum_{m=1}^M \left\|\mathbf{y}_m - \mathbf{D}_m\mathbf{x}_m\right\|_2^2 + \lambda\left\|\mathbf{x}_m\right\|_1 + \mu w_m^* \left\|\mathbf{x}_m - \overline{\mathbf{x}}^*\right\|_2^2 \tag{7}$$

It is easy to see that the objective function (7) can be viewed as a sum of M individual functions, i.e., $J^1\left(\mathbf{x}_m |_{m=1}^M\right) = \sum_{m=1}^M J^{1,m}\left(\mathbf{x}_m\right)$, where

$$\begin{aligned} J^{1,m}\left(\mathbf{x}_m\right) &= \left\|\mathbf{y}_m - \mathbf{D}_m\mathbf{x}_m\right\|_2^2 + \lambda\left\|\mathbf{x}_m\right\|_1 + \mu w_m^* \left\|\mathbf{x}_m - \overline{\mathbf{x}}^*\right\|_2^2 \\ &= \left\|\mathbf{y}_m - \mathbf{D}_m\mathbf{x}_m\right\|_2^2 + \lambda\left\|\mathbf{x}_m\right\|_1 + \left\|\sqrt{\mu w_m^*}\,\overline{\mathbf{x}}^* - \sqrt{\mu w_m^*}\,\mathbf{I}\mathbf{x}_m\right\|_2^2 \\ &= \left\|\begin{bmatrix}\mathbf{y}_m \\ \sqrt{\mu w_m^*}\,\overline{\mathbf{x}}^*\end{bmatrix} - \begin{bmatrix}\mathbf{D}_m \\ \sqrt{\mu w_m^*}\,\mathbf{I}\end{bmatrix}\mathbf{x}_m\right\|_2^2 + \lambda\left\|\mathbf{x}_m\right\|_1 \\ &= \left\|\mathbf{y}'_m - \mathbf{D}'_m\mathbf{x}_m\right\|_2^2 + \lambda\left\|\mathbf{x}_m\right\|_1 \end{aligned} \tag{8}$$

where $\mathbf{y}'_m = \left[\left(\mathbf{y}_m\right)^T, \left(\sqrt{\mu w_m^*}\,\overline{\mathbf{x}}^*\right)\right]^T$, $\mathbf{D}'_m = \left[\left(\mathbf{D}_m\right)^T, \left(\sqrt{\mu w_m^*}\,\mathbf{I}\right)^T\right]^T$, $\mathbf{I}$ is an indentify matrix. From equation (7), we can see that the optimization problem (6) can be modified as M sub-optimization problems, i.e., $\mathbf{x}_m^* = \arg\min_{\mathbf{x}_m} J^{1,m}\left(\mathbf{x}_m\right), m = 1,2,\ldots,M$, each of which is a standard LASSO problem that can be optimized by using the SPAMS (SPArse Modeling Software) package (http://spams-devel.gforge.inria.fr/).

Proposition 2: Given the optimal solutions $\mathbf{x}_m^*\big|_{m=1}^M$ and $\mathbf{w}^*$, the optimal average coefficient vector $\overline{\mathbf{x}}^*$ can be solved by a simple weighted average operator.

If the optimal values $\mathbf{x}_m^*\big|_{m=1}^M$ and $\mathbf{w}^*$ are given, the optimal average coefficient vector can be obtained by solving the following problem,

$$\overline{\mathbf{x}}^* = \arg\min_{\overline{\mathbf{x}}} J^2\left(\overline{\mathbf{x}}\right) \tag{9}$$

where the objective function $J^2\left(\overline{\mathbf{x}}\right)$ is defined as $J^2\left(\overline{\mathbf{x}}\right) = \sum_{m=1}^M \mu w_m^* \left\|\mathbf{x}_m^* - \overline{\mathbf{x}}\right\|_2^2$. This problem is a standard least squares problem and its closed-from solution can be obtained by setting the derivation $\partial J^2\left(\overline{\mathbf{x}}\right) \big/ \partial\overline{\mathbf{x}}$ to zero.

$$\frac{\partial J^2\left(\bar{\mathbf{x}}\right)}{\partial\bar{\mathbf{x}}} = 2\mu\sum_{m=1}^{M} w_m^*\left(\bar{\mathbf{x}} - \mathbf{x}_m^*\right) = 2\mu\left[\left(\sum_{m=1}^{M} w_m^*\right)\bar{\mathbf{x}} - \sum_{m=1}^{M} w_m^*\mathbf{x}_m^*\right] = \mathbf{0} \tag{10}$$

So, the optimal solution of the average coefficient vector can be obtained by $\bar{\mathbf{x}}^* = \sum_{m=1}^{M} w_m^*\mathbf{x}_m^* \Big/ \sum_{m=1}^{M} w_m^*$. Due to the non-negativity of the coefficient vectors $\mathbf{x}_m^*\big|_{m=1}^{M}$, the optimal average vector $\bar{\mathbf{x}}^*$ is also negative.

Proposition 3: Given the optimal solutions $\mathbf{x}_m^*\big|_{m=1}^{M}$ and $\bar{\mathbf{x}}^*$, the optimal weight vector $\mathbf{w}^*$ can be obtained by M product operators separately.

If the optimal values $\mathbf{x}_m^*\big|_{m=1}^{M}$ and $\bar{\mathbf{x}}^*$ are given, the optimization problem (4) can be converted into the minimization problem (11).

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} J^3\left(\mathbf{w}\right) \tag{11}$$

in which the objective function $J^3\left(\mathbf{w}\right) = \sum_{m=1}^{M} J^{3,m}\left(\mathbf{w}\right)$ that is a sum of M sub-equations $J^{3,m}\left(w_m\right) = \mu w_m \left\|\mathbf{x}_m - \bar{\mathbf{x}}\right\|_2^2 + \eta w_m \log\frac{w_m}{w'_m}$. Thus, the optimal weight vector $\mathbf{w}^*$ can be obtained by solving M sub-problems $w_m^* = \arg\min_{w_m} J^{3,m}\left(w_m\right),\ j = 1,2,...,M$. By setting the derivation $\partial J^{3,m}\left(w_m\right)\big/\partial w_m$ to zero, the optimal value w_m^* can be obtained.

$$\begin{aligned}\frac{J^{3,m}\left(w_m\right)}{\partial w_m} &= \frac{\partial\left\{\mu w_m\left\|\mathbf{x}_m^* - \bar{\mathbf{x}}^*\right\|_2^2 + \eta w_m\log\frac{w_m}{w'_m}\right\}}{\partial w_m}\\ &= \mu\left\|\mathbf{x}_m^* - \bar{\mathbf{x}}^*\right\|_2^2 + \eta\log w_m + \eta - \eta\log w'_m\\ &= 0\end{aligned} \tag{12}$$

Thus, we can obtain $w_m^* = \exp\left(-1 - \frac{\mu}{\eta}\left\|\mathbf{x}_m^* - \bar{\mathbf{x}}^*\right\|_2^2\right) w'_m$, the physical meaning of which is very intuitive. The former component $\exp\left(-1 - \frac{\mu}{\eta}\left\|\mathbf{x}_m^* - \bar{\mathbf{x}}^*\right\|_2^2\right)$ measures the inconsistency between the coding coefficient vector $\mathbf{x}_m^*$ and the average coding vector $\bar{\mathbf{x}}^*$; and the later one makes the solution be similar with the reference weight vector.

Table 1. The iteration algorithm for solving the optimization problem (4)

Input: The observation patches $\mathbf{y}_m\big|_{m=1}^{M}$, the candidate dictionaries $\mathbf{D}_m\big|_{m=1}^{M}$, the reference weight vector $\mathbf{w}'$, and the regularization parameters λ, μ and η.
1: Initialize the average coefficient vector $\bar{\mathbf{x}}^{0} = \mathbf{0}$ and weight vector $\mathbf{w}^{0} = \mathbf{w}'$
2: Iterate
3: Obtain $\mathbf{x}_m^i\big\|_{m=1}^{M}$ by solving $\mathbf{x}_m^i\big\|_{m=1}^{M} = \arg\min_{\mathbf{x}_m\|_{m=1}^{M}} \sum_{m=1}^{M} \left\Vert\mathbf{y}_m - \mathbf{D}_m\mathbf{x}_m\right\Vert_2^2 + \lambda\left\Vert\mathbf{x}_m\right\Vert_1 + \mu w_m^{i-1}\left\Vert\mathbf{x}_m - \bar{\mathbf{x}}^{i-1}\right\Vert_2^2$ by using the LASSO method (based on **Proposition 1**).
4: Calculate the average coefficient vector: $\bar{\mathbf{x}}^{i} = \sum_{m=1}^{M} w_m^{i-1}\mathbf{x}_m^i \Big/ \sum_{m=1}^{M} w_m^{i-1}$.
5: Update the weight vector: $w_m^i = \exp\left(-1 - \frac{\mu}{\eta}\left\Vert\mathbf{x}_m^i - \bar{\mathbf{x}}^{i}\right\Vert_2^2\right) w_m'$.
6: Until Convergence or termination
Output: The optimal vectors $\mathbf{x}_m^*\big\|_{m=1}^{M}$, $\bar{\mathbf{x}}^{*}$ and $\mathbf{w}^{*}$.

By the above-mentioned three propositions, the optimization problem (4) can be solved iteratively. The iteration algorithm for solving the optimization problem (4) is presented in Table 1. The iteration operations will be terminated when a stopping criterion is met, e.g., the difference of the average coefficient vector ($\left\Vert\bar{\mathbf{x}}^{i} - \bar{\mathbf{x}}^{i-1}\right\Vert_2 \le 0.01$) or a maximal number of iteration steps.

2.4 The tracking framework

Based on the proposed relaxed sparse representation model, we develop a tracking algorithm by using the Bayesian inference framework. In general, object tracking can be casted as a Bayesian inference problem in a hidden Markov model. Given continuous observation image patches $\mathrm{D}^t = \{\mathbf{d}^1, \mathbf{d}^2, ..., \mathbf{d}^t\}$ up to the t-th frame, the aim is to infer the hidden state variable $\mathbf{z}^t$ recursively, i.e.,

$$p(\mathbf{z}_t \mid \mathrm{D}_t) \propto p(\mathbf{d}_t \mid \mathbf{z}_t) \int p(\mathbf{z}_t \mid \mathbf{z}_{t-1})\, p(\mathbf{z}_{t-1} \mid \mathrm{D}_t)\, d\mathbf{z}_{t-1} \tag{13}$$

where $p(\mathbf{z}_t \mid \mathbf{z}_{t-1})$ stands for the motion model between two consecutive frames and $p(\mathbf{d}_t \mid \mathbf{z}_t)$ is the observation model that estimates the likelihood function for each candidate. The overall framework of the tracking method has been illustrated in Fig. 1. Similar to [16], the affine transform with six parameters is adopted to depict the motion model $p(\mathbf{z}_t \mid \mathbf{z}_{t-1})$, in which $\mathbf{x}_t = \{x_t, y_t, \theta_t, s_t, \alpha_t, \phi_t\}$ denote the x, y translations, rotation angle, scale, aspect ratio, and skew in the t-th frame. Then the random walk process are adopted to describe the state transition, i.e., $p(\mathbf{z}_t \mid \mathbf{z}_{t-1}) = \mathrm{N}\left(\mathbf{z}_t; \mathbf{z}_{t-1}, \boldsymbol{\psi}\right)$, where $\boldsymbol{\psi} = \{\sigma_x^2, \sigma_y^2, \sigma_\theta^2, \sigma_s^2, \sigma_\alpha^2, \sigma_\phi^2\}$ is a diagonal covariance matrix.

In the t-th frame, we solve the following optimization problem,

$$\min_{\bar{\mathbf{x}}^t, \mathbf{x}_m^t|_{m=1}^M, \mathbf{w}^t} \sum_{m=1}^{M} \left\| \mathbf{y}_m - \mathbf{D}_m^t \mathbf{x}_m^t \right\|_2^2 + \lambda \sum_{m=1}^{M} \left\| \mathbf{x}_m^t \right\|_1 + \mu \sum_{m=1}^{M} w_m^t \left\| \mathbf{x}_m^t - \bar{\mathbf{x}}^t \right\|_2^2 + \eta \sum_{m=1}^{M} w_m^t \log \frac{w_m^t}{w_m^{t-1}} \quad (13)$$

where $\mathbf{y}_m|_{m=1}^M$ denotes the template, $\mathbf{D}_m^t = \left\{ \left[\mathbf{d}_m^t \right]_1, \left[\mathbf{d}_m^t \right]_2, ..., \left[\mathbf{d}_m^t \right]_N \right\}$ stands for the m -th sub-dictionary related with N candidate states, $\mathbf{w}^{t-1} = \left[w_1^{t-1}, w_2^{t-1}, ..., w_M^{t-1} \right]$ is the weight vector of the $(t-1)$-th frame. After solving equation (13) by the iteration algorithm in **Table 1**, the likelihood of each candidate can be measured by $p\left(\mathbf{d}_t^i \mid \mathbf{z}_t^i\right) = \left(\bar{\mathbf{x}}^*\right)_i$, $i = 1, 2, ..., N$. Then the optimal state $\mathbf{z}_t^*$ inferred by the Bayesian framework. We also note that the reference weight vector is initialized as $w_t^i = \frac{1}{M}, i = 1, ..., M$ and then is updated frame by frame.

During the tracking process, it is necessary to update the template of the tracked object to capture the appearance changes. After obtaining the optimal state $\mathbf{z}^*$, we extract its corresponding image patch $\mathbf{d}^*$ and update the target template in a fragment-based manner, i.e.,

$$\begin{cases} \mathbf{y}_m \leftarrow (1-\eta)\mathbf{y}_m + \eta \mathbf{d}_m^*, if \ \left\| \mathbf{y}_m - \mathbf{d}_m^* \right\|_2 < \varepsilon \\ \mathbf{y}_m \leftarrow \mathbf{y}_m, \qquad otherwise \end{cases} \quad (14)$$

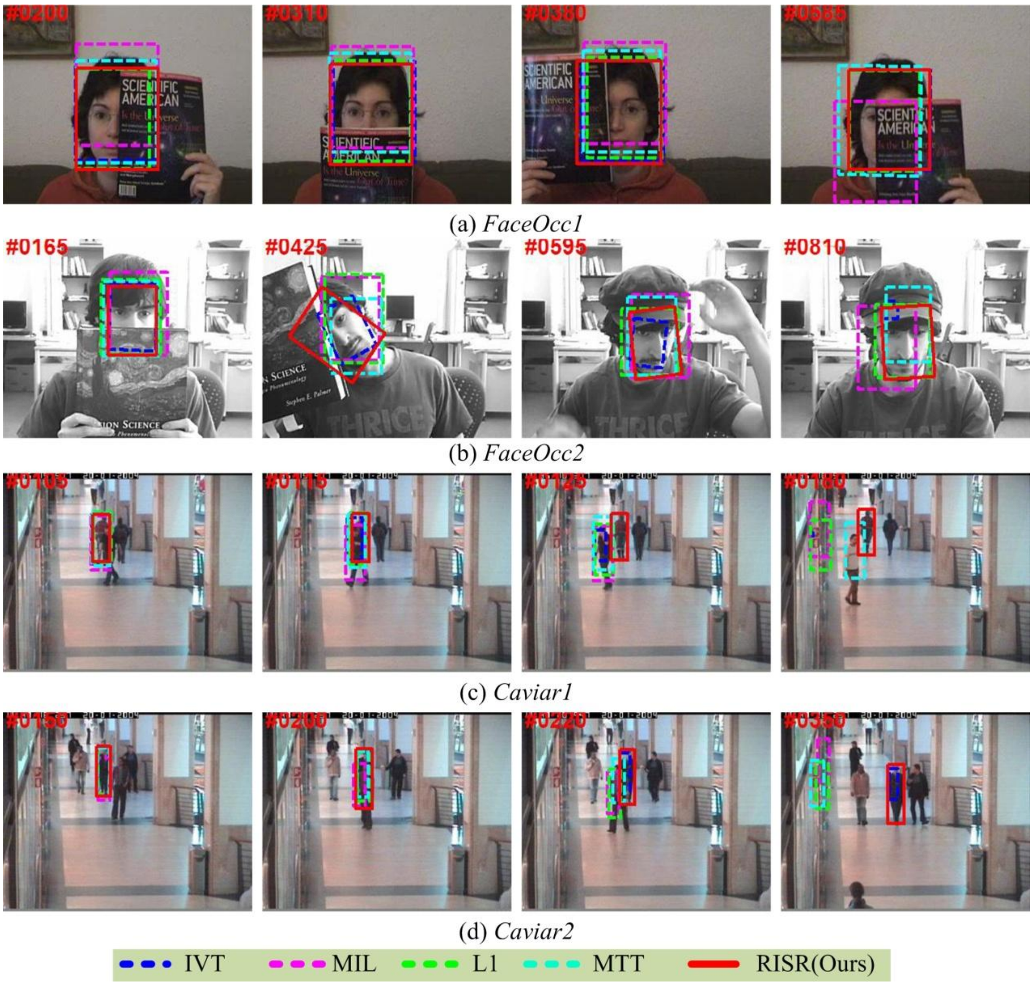
in which $\eta = 0.95$ is a update rate and $\varepsilon = 0.1$ is a predefined threshold.

3. Experiments

In this paper, we implement our tracker in the MATLAB platform, which runs 18 frames per second on a PC machine with Intel i5-M560 CPU (2.67 GHz) with 2 GB memory. For each image sequence, the bounding box of the tracked object is manually labeled in the first frame for initializing our tracker. The affine parameters are set to be $\boldsymbol{\psi} = diag\left(4, 4, 1e^{-2}, 5e^{-3}, 1e^{-3}, 1e^{-3}\right)$ for sampling candidates in each frame. To model the appearance feature of each patch, we firstly resize each patch to 32×32 pixels and then divide it into $4 \times 4 (=16)$ fragments. 600 particles are used to balance the effectiveness and efficiency. The parameter for sparse regularization λ is set as 0.1, and other parameters are set to be $\mu = 0.01$ and $\eta = 0.01$.

We adopt many challenging video clips to evaluate the proposed tracker in comparison with ten recent trackers, including five sparisty-based tracking algorithms (accelerated proximal gradient L1 (L1) [21], multitask tracking (MTT) [24], local sparse appearance tracking (LSAT) [29], two view sparse represenation (TVSR) [35]) and other six methods (fragment-based tracking (Frag) [30], incremental visual tracking (IVT) [15], multiple instance learning (MIL) [12], visual tracking decomposition (VTD) [16], tracking-learning-detection (TLD) [13] and probability continuous outlier model (PCOM) [34]). The challenging factors of these video clips include partial occlusion, illumination variation, scale change, pose change, background clutter and so on. Both qualitative and quantitative results are presented as follows.

3.1 Qualitative Results



(a) *FaceOcc1*

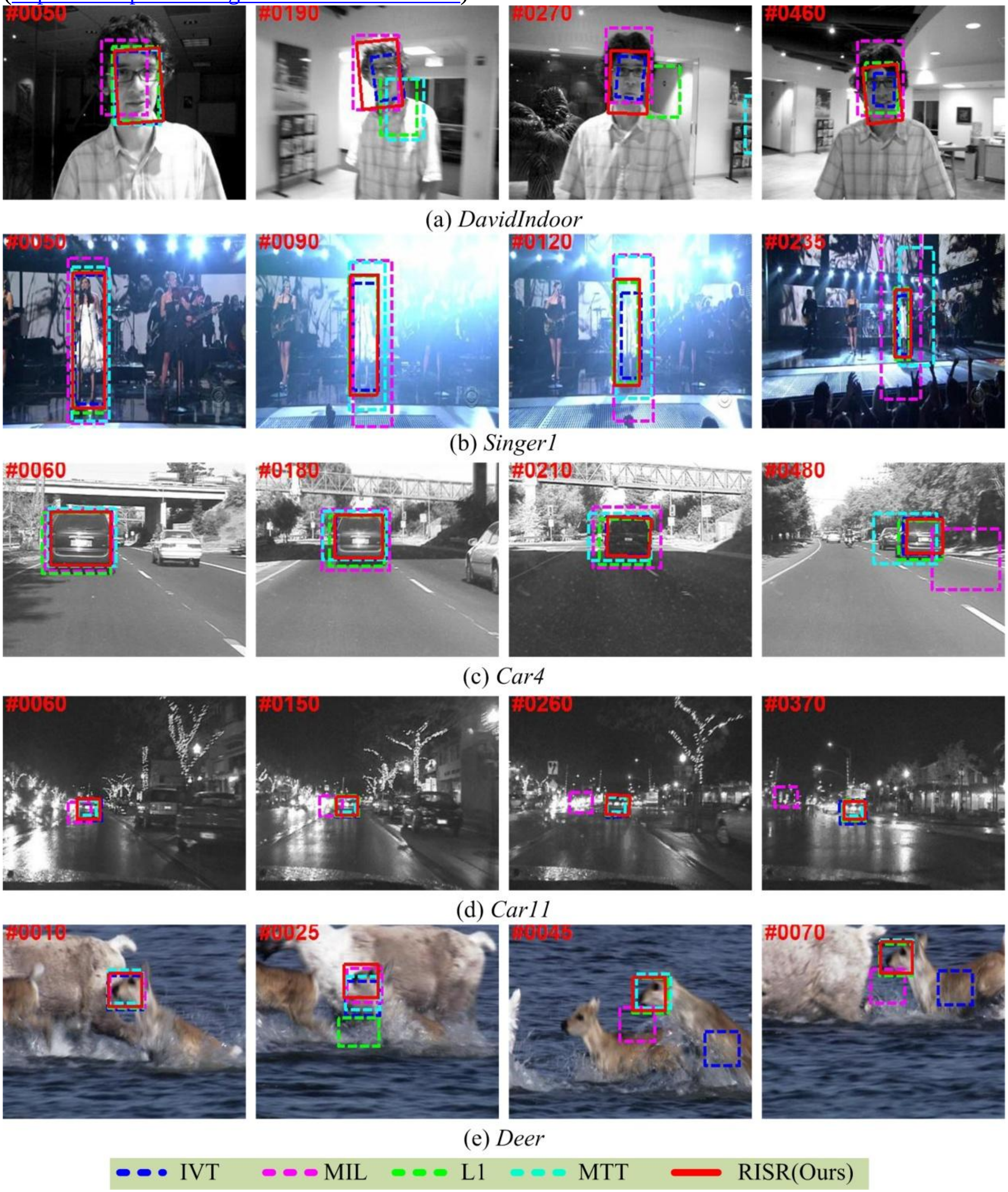
(b) *FaceOcc2*

(c) *Caviar1*

(d) *Caviar2*

Fig. 2. An illustration of selected tracking results of different trackers in partial occlusions.

Fig. 2 and **Fig. 3** provide some qualitative results to compare our tracker with other four tracking methods, including two baseline algorithms (IVT and MIL) and two sparse ones (L1 and MTT). In **Fig. 2**, our tracker is compared with other tracking methods when partial occlusions occur during the tracking process. It can be seen from this figure that our tracker performs better than other algorithms in handling these cases due to the fragment-based scheme and relaxed sparse representation. The IVT method is very sensitive to partial occlusions as its adopted PCA representation model cannot depict outliers (e.g., *Occlusion2* and *Caviar1*). The MIL method uses the multiple instance learning technique to deal with the ambiguity of positive samples, however this algorithm cannot handle the case when the tracked object occluded by other objects with similar appearance (e.g., *Caviar1* and *Caviar2*). Although the FragT method also uses the fragment-based object representation to handle partial occlusions, it performs poorly in more complex conditions (e.g., *Occlusion2* and *Caviar2*). Both the L1 and MTT trackers are motivated by sparse representation, which adopts a series of trivial templates to model outliers (i.e., partial occlusions) explicitly. But they also achieve not good performance as the traditional sparse representation model cannot effectively model the relationships among candidates (e.g., *Caviar1* and *Caviar2*). Besides, **Fig. 3** show representative results on other five challenging videos. It can be seen from this figure that the proposed tracking algorithm achieves good performance in dealing with pose variation (*DavidIndoor*), illumination change (*DavidIndoor*, *Singer1*, *Car4*) and cluttered background (*Car11* and *Deer*). In addition, Fig.4 demonstrates the tracking results of different algorithms on two challenging image sequences in the PETS dataset (http://www-prima.imag.fr/PETS04/index.html).



(a) *DavidIndoor*

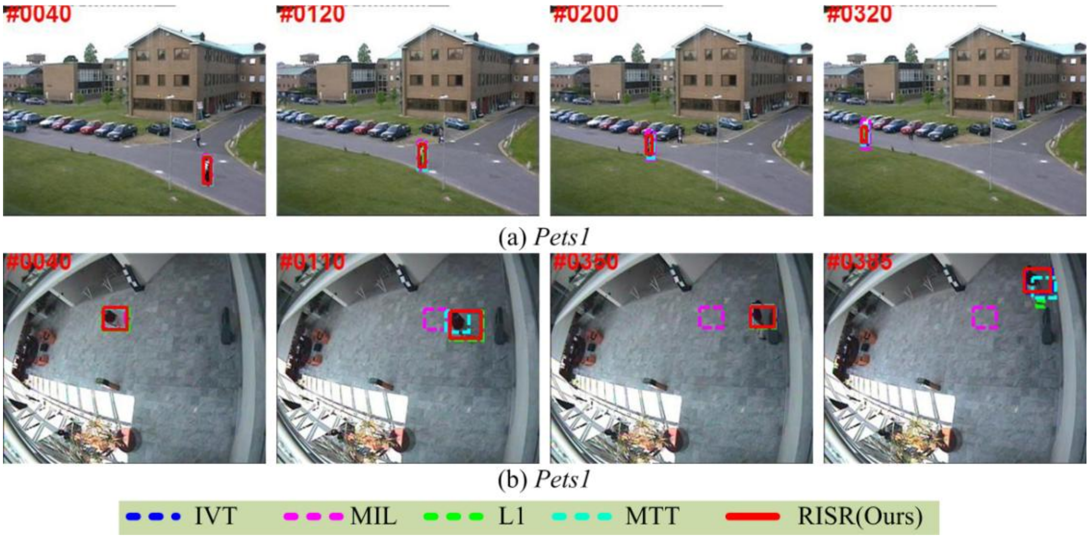
(b) *Singer1*

(c) *Car4*

(d) *Car11*

(e) *Deer*

Fig. 3. An illustration of selected tracking results of different trackers in other conditions.



(a) *Pets1*

(b) *Pets1*

Fig. 4. An illustration of selected tracking results of different trackers in two challenging sequences from the PETS dataset.

3.2 Quantitative Results

To evaluate our tracker and other trackers quantitatively, we use two popular criteria, i.e., center location error (CLE) and F-meansure (F). The center location error is defined as the Euclidean distance between the center location of the ground truth and the center location obtained by a tracker. It is obvious that a good tracker intends to obtain small CE values in the test sequences. However, this rule does not consider scale and rotation changes. In addition, we also adopt the F-meansure (F) [5] to further evaluate different trackers (the F rule is defined in equation (15)).

$$F = \frac{2 \times Pr(R_T, R_G) \times Re(R_T, R_G)}{Pr(R_T, R_G) + Re(R_T, R_G)} \tag{15}$$

in which R_G and R_T are ground truth and the tracked bounding boxes respectively, $Pr(R_T, R_G) = area(R_T \cap R_G)/area(R_G)$, $Re(R_T, R_G) = area(R_T \cap R_G)/area(R_T)$, and $area(X)$ denotes the area of the region X. **Table 2** and **Table 3** report the average CLE and F values for different tracking algorithms on the test image sequences, from which we can see that the proposed RISR tracking method achieves better performance than other state-of-the-art trackers.

Table 2. Average center location errors (ACLE) of different trackers.

Method Video	FragT [30]	IVT [15]	MIL [12]	VTD [16]	TLD [13]	L1 [21]	LAST [29]	MTT [24]	PCOM [34]	TVSR [35]	RISR (Ours)
FaceOcc1	5.6	9.2	32.3	11.1	17.6	6.8	5.3	14.1	5.9	4.4	3.2
FaceOcc2	15.5	10.2	14.1	10.4	18.6	6.3	58.6	9.2	4.0	5.3	4.8
Caviar1	5.7	45.2	48.5	3.9	5.6	50.1	1.8	20.9	1.4	1.5	0.9
Caviar2	5.6	8.6	70.3	4.7	8.5	63.1	45.6	65.4	1.8	2.8	2.5
DavidIndoor	76.7	3.6	16.1	13.6	9.7	14.3	4.9	124.0	3.8	3.1	5.0
Singer1	22.0	8.5	15.2	4.1	32.7	3.1	14.5	41.2	5.2	3.4	3.7
Car4	179.8	2.9	60.1	12.3	18.8	16.4	3.3	37.2	4.6	7.3	3.5
Car11	63.9	2.1	43.5	27.1	25.1	1.7	4.1	1.8	2.2	3.1	1.8
Deer	92.1	127.5	66.5	11.9	25.7	38.4	69.8	9.2	13.9	25.8	10.0

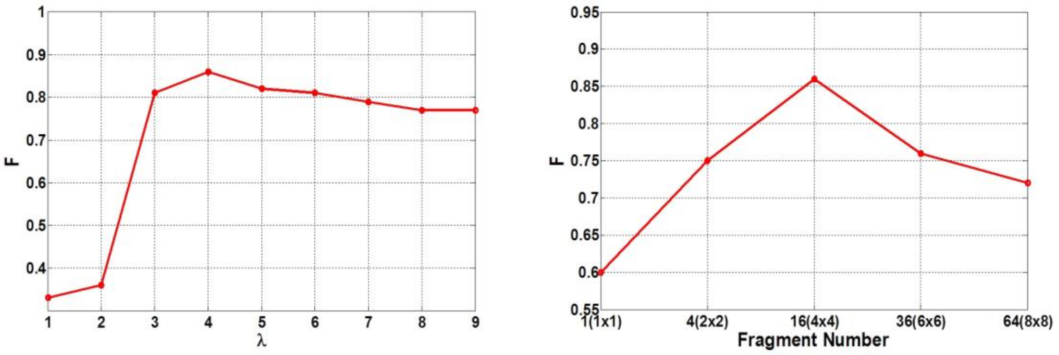
Pets1	11.4	1.9	3.3	2.9	9.9	2.2	22.0	3.7	2.2	2.5	2.0
Pest2	54.8	6.0	15.4	18.6	13.2	60.7	34.9	9.8	8.4	6.2	5.8
Average	48.5	20.5	35.0	11.0	16.9	23.9	24.1	30.6	4.9	5.9	3.9

Table 3. Average overlap rates (AOR) of different trackers.

Method Video	FragT [30]	IVT [15]	MIL [12]	VTD [16]	TLD [13]	L1 [21]	LAST [29]	MTT [24]	PCOM [34]	TVSR [35]	RISR (Ours)
FaceOcc1	0.95	0.91	0.73	0.87	0.77	0.93	0.95	0.88	0.92	0.92	0.95
FaceOcc2	0.74	0.69	0.75	0.71	0.63	0.82	0.39	0.83	0.85	0.80	0.83
Caviar1	0.80	0.29	0.28	0.91	0.82	0.30	0.92	0.55	0.92	0.91	0.93
Caviar2	0.70	0.58	0.32	0.79	0.78	0.36	0.34	0.38	0.78	0.84	0.87
DavidIndoor	0.27	0.82	0.61	0.68	0.74	0.69	0.76	0.33	0.82	0.88	0.87
Singer1	0.46	0.79	0.46	0.88	0.54	0.91	0.64	0.43	0.75	0.78	0.91
Car4	0.28	0.96	0.45	0.84	0.77	0.82	0.95	0.67	0.84	0.88	0.95
Car11	0.11	0.89	0.24	0.50	0.48	0.91	0.65	0.73	0.82	0.86	0.89
Deer	0.11	0.26	0.30	0.72	0.50	0.57	0.45	0.74	0.59	0.35	0.77
Pets1	0.55	0.75	0.56	0.71	0.57	0.68	0.39	0.77	0.76	0.74	0.78
Pets2	0.10	0.76	0.49	0.51	0.60	0.38	0.49	0.62	0.65	0.72	0.75
Average	0.46	0.70	0.47	0.74	0.66	0.67	0.63	0.63	0.79	0.78	0.86

3.3 The effects of parameters and components

In this subsection, we investigate the effects of two critical parameters, the sparsity regularization parameter λ and the fragment number M . First, the choice of parameter λ is a critical parameter that controls the sparsity level. **Fig. 5 (a)** demonstrates the tracking performance (i.e., F-measure) with different λ values. If λ is too small, the solution will be too trivial and not sparse, which will introduce too much noise in inferring the similarities of different candidates. On the other hand, if λ is too large, the sparisty will be over-emphasized, which may lead to select a very small number of candidates. Thus, the tracking performance is also not good. Second, the number of fragments is also very important to our tracker (the performance of different fragments is shown in **Fig. 5 (b)**). If the fragment number is very small, the tracker cannot achieve good performance as it is not able to model outliers (such as partial occlusions or local illumination changes) effectively by using a very small number of fragments. For another, if the fragment number is too large, the size of each fragment will be too small to capture sufficient visual information, which will also leads the tracker's performance is not satisfying. Based on the reported results in **Fig. 5**, we set $\lambda = 0.1$ and $M = 16$ as default values in this work.



(a) The effect of sparisty regularization parameter

(b) The effect of fragment number

Fig. 5. The effects of critical parameters.

In addition, Fig. 6 show the effects of different components, in which L1 denotes the original L1 tracker [21], ISR denotes the tracking method based on inverse sparse representation process without using the fragment scheme, FISR is the fragment-based ISR tracker, and RISR indicates the final tracking method that combines the inverse sparse representation, fragment scheme and adaptive weight scheme within a unified framework. From this figure, we can see that both fragment and adaptive weight schemes facilitate the improvements of tracking performance.

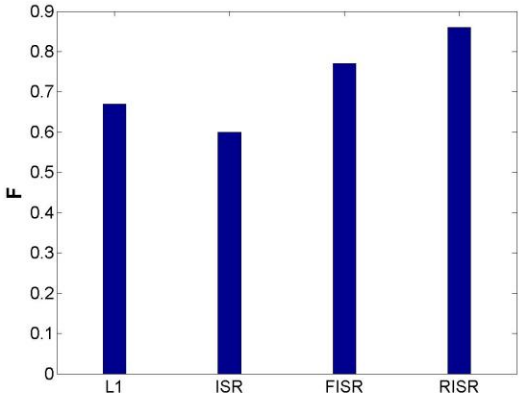


Fig. 6. The effects of different components.

4. Conclusion

In this work, we develop a novel online object tracking algorithm based on our relaxed inverse sparse representation algorithm. First, we treat the tracking problem as an inverse sparse representation process, in which a given template of the tracked object can be sparsely represented by the candidate samples in each frame. In addition, we introduce a relaxed constraint term to make the inverse sparse representation process be more flexible and a weight prior as a reference of the relaxed term. Then, a novel tracking method is designed based on the proposed representation model and a simple online update manner within the Bayesian framework. Last but not least, we conduct many experiments to compare our tracker with other recent trackers. The results show that our tracker performs better than other compared tracking algorithms.

References

[1] Hanxuan Yang, Ling Shao, Feng Zheng, Liang Wang and Zhan Song, “Recent advances and trends in visual tracking: A review,” *Neurocomputing*, vol.74, no.18, pp.3823-3831, November , 2011. Article (CrossRef Link)

[2] Xi Li, Weiming Hu, Chunhua Shen, Zhongfei Zhang, Anthony R. Dick and Anton van den Hengel, “A survey of appearance models in visual object tracking,” A*CM Transactions on Intelligent Systems and Technology s*, vol. 4, no. 4, pp. 58, December, 2013. Article (CrossRef Link)

[3] Michael D. Breitenstein, Fabian Reichlin and Bastian Leibe, “Online multiperson tracking-by-detection from a single, uncalibrated camera,” *IEEE Transcations on Pattern Analysis and Machine Intellignece*, vol. 23, no. 9, pp. 1820-1833, Septermber, 2011. Article (CrossRef Link)

[4] Weizhi Nie, Anan Liu, Yuting Su, Huanbo Luan, Zhaoxuan Yang, Liujuan Cao and Rongrong Ji, “Single/cross-camera multiple-person tracking by graph matchin,g,” *Neurocomputing* ,vol. 139,

pp. 220-232, September, 2014. Article (CrossRef Link)

[5] Yue Gao, Rongrong Ji, Longfei Zhang and Alexander Hauptmann, "Symbiotic tracker ensemble toward a unified tracking framework," *IEEE Transcations on Circulits and Systems for Video Technology*, vol. 24, no. 7, July, 2014. Article (CrossRef Link)

[6] Yi Wu, Jongwoo Lim and Ming-Hsuan Yang, "Online object tracking: a benchmark," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2411–2013, June 23-28, 2013. Article (CrossRef Link)

[7] Dorin Comaniciu, Visvanathan Ramesh and Peter Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.25, no.5, pp.564-575, May, 2003. Article (CrossRef Link)

[8] P. P´erez, C. Hue, J. Vermaak and M. Gangnet, "Color-based probabilistic tracking," in *Proc. of European Conference on Computer Vision*, pp.661-675, May 28–31, 2002. Article (CrossRef Link)

[9] Yuan Li, Haizhou Ai, Takayoshi Yamashita, Shihong Lao and Masato Kawade, "Tracking in Low Frame Rate Video: A cascade particle filter with discriminative observers of different life spans," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1728–1740, October, 2008. Article (CrossRef Link)

[10] Shai Avidan, "Support vector tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp.1064-1072, September, 2004. Article (CrossRef Link)

[11] Shai Avidan, "Ensemble tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp.261-271, January, 2007. Article (CrossRef Link)

[12] Helmut Grabner and Horst Bischof, "On-line boosting and vision," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 260-267, June 17-22, 2006. Article (CrossRef Link)

[13] Dong Wang and Huchuan Lu, "Fast and effective color-based object tracking by boosted color distribution," *Pattern analysis and application*, vol. 16, pp. 647-661, July, 2013. Article (CrossRef Link)

[14] Helmut Grabner, C. Leistner, and Horst Bischof, "Semi-supervised on-line boosting for robust tracking," in *Proc. of European Conference on Computer Vision*, pp.234-247, October 12-18, 2008. Article (CrossRef Link)

[15] Boris Babenko, Ming-Hsuan Yang and Serge Belongie, "Robust Object Tracking with Online Multiple Instance Learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.33, no.8, pp.1619-1632, August, 2011. Article (CrossRef Link)

[16] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas, "Tracking-learning-detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409-1422, July, 2012. Article (CrossRef Link)

[17] Simon Baker and Iain Matthews, "Lucas-Kanade 20 years on: a unifying framework," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, February, 2004. Article (CrossRef Link)

[18] David A. Ross, Jongwoo Lim, Ruei-Sung Lin and Ming-Hsuan Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 125–141, August, 2008. Article (CrossRef Link)

[19] Junseok Kwon and Kyoung Mu Lee, "Visual tracking decomposition," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1269–1276, June 13-18, 2010. Article (CrossRef Link)

[20] John Wright, Yi Ma, Julien Mairal, Guillermo Sapiro, Thomas S. Huang and Shuicheng Yan, "Sparse Representation for Computer Vision and Pattern Recognition," *Proceedings of the IEEE*, vol.98, no.6, pp.1031-1044, June, 2010. Article (CrossRef Link)

[21] John Wright, Allen Y. Yang, Arvind Ganesh, S.Shankar Sastry and Yi Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210-227, February, 2009. Article (CrossRef Link)

[22] Xue Mei and Haibin Ling, "Robust visual tracking using L1 minimization," in *Proc. of*

International Conference on Computer Vision, pp. 1436-1443, September 29-October 2, 2009. Article (CrossRef Link)

[23] Xue Mei, Haibin Ling, Yi Wu, Erik Blasch and Li Bai, “Minimum error bounded efficient L1 tracker with occlusion detection,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1257-1264, June 20-25, 2011. Article (CrossRef Link)

[24] Chenglong Bao, Yi Wu, Haibin Ling and Hui Ji , “Real time robust L1 tracker using accelerated proximal gradient approach,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1830-1837, June 16-21, 2012. Article (CrossRef Link)

[25] Wei Zhong, Huchuan Lu and Minghsuan Yang, “Robust object tracking via sparse collaborative appearance model,” *IEEE Transaction on Image Processing*, vol. 23, no. 5, pp. 2356-2368, May, 2014. Article (CrossRef Link)

[26] Bohan Zhuang, Huchuan Lu, Ziyang Xiao and Dong Wang, “Visual tracking via discriminative sparse similarity map,” *IEEE Transaction on Image Processing*, vol. 23, no. 4, pp.1872-1881, April, 2014. Article (CrossRef Link)

[27] Tianzhu Zhang, Bernard Ghanem, Si Liu and Narendra Ahuja, “Robust visual tracking via multi-task sparse learning,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2042-2049, June 16-21, 2012. Article (CrossRef Link)

[28] Dong Wang, Huchuan Lu and Minghsuan Yang, “Online object tracking with sparse prototypes,” *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 314-325, January, 2013. Article (CrossRef Link)

[29] Dong Wang, Huchuan Lu, “On-line learning parts-based representation via incremental orthogonal projective non-negative matrix factorization,” *Signal Processing*, vol. 93, no. 6, pp. 1608-1623, June, 2013. Article (CrossRef Link)

[30] Dong Wang, Huchuan Lu, Ziyang Xiao and Minghsuan Yang, “Inverse sparse tracker with a locally weighted distance metric”, *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2646-2657, September, 2015. Article (CrossRef Link)

[31] Meng Yang, Lei Zhang, David Zhang and Shenlong Wang, “Relaxed collaborative representation for pattern classification,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2224-2231, June 16-21, 2012. Article (CrossRef Link)

[32] Amit Adam, Ehud Rivlin and Ilan Shimshoni, “Robust fragments-based tracking using the integral histogram,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 798–805, June 17-22, 2006. Article (CrossRef Link)

[33] Baiyang Liu, Junzhou Huang, Lin Yang and Casimir Kulikowsk, “Robust tracking using local sparse appearance model and K-selection,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1313-1320, June 20-25, 2011. Article (CrossRef Link)

[34] Dong Wang and Huchuan Lu, “Visual tracking via probability continuous outlier model”, in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3478-3485, June 24-27, 2014. Article (CrossRef Link)

[35] Dong Wang, Huchuan Lu and Chunjuan Bo, “Online visual tracking via two view sparse representation”, *IEEE Signal Processing Letters*, vol. 21, no.9, September, 2014. Article (CrossRef Link)

Junxing Zhang received the B.E. degree and M.S. degree in Electronic Information Engineering, Shandong University, China, in 1992 and 1995 respectively. He also received the Ph.D degree in detection technology and automation devices, Northeastern University, China, in 1998. He is currently a faculty in the College of Information and Communication Engineering of Dalian Nationalities University. His research interests include pattern recognition, speech processing and so on.

Chunjuan Bo received the B.E. degree in Electronic Information Engineering, Dalian Nationalities University, China, in 2008. She also received M.S. degree in Communication and Information System, Dalian University of Technology, China, in 2010. She is currently a faculty in the College of Electromechanical Engineering of Dalian Nationalities University. Her research interests include image classification, pattern recognition and so on.

Jianbo Tang received the B.E. degree in Hoisting and Conveying and Engineering Machinery, Dalian University of Technology, China, in 1987. He is currently a faculty in the College of Electromechanical Engineering of Dalian Nationalities University. His research interests include design of precision machinery, mechanical electronic and so on.

Peng Song received the B.E. degree in Electronic Engineering, Minzu University of China, in 2000. She also received M.S. degree in Power Machinery Engineering, Dalian University of Technology, China, in 2007. She is currently a faculty in the College of Electromechanical Engineering of Dalian Nationalities University. Her research interests include Control theory, Electric vehicle drive control and so on.