A Simplified Low Rank and Sparse Model for Visual Tracking

Mi Wang, Huaxin Xiao, Yu Liu, Wei Xu and Maojun Zhang

Institute of Information Systems and Management, National University of Defense,
No. 109 Deya Road Kaifu District Changsha City Hunan Prov., 410073, Changsha, China
{coolsa007, huaxin_xiao}@163.com, jasonyuliu@nudt.edu.cn, {545406612, 1023133126}@qq.com

Keywords: Visual Tracking, Sparse and Low-Rank Representation.

Abstract: Object tracking is the process of determining the states of a target in consecutive video frames based on properties of motion and appearance consistency. Numerous tracking methods using low-rank and sparse constraints perform well in visual tracking. However, these methods cannot reasonably balance the two characteristics. Sparsity always pursues a sparse enough solution that ignores the low-rank structure and vice versa. Therefore, this paper replaces the low-rank and sparse constraints with $\| \cdot \|_2$ norm. A simplified low-rank and sparse model for visual tracking (LRSVT), which is built upon the particle filter framework, is proposed in this paper. The proposed method first prunes particles which are different with the object and selects candidate particles for efficiency. A dictionary is then constructed to represent the candidate particles. The proposed LRSVT algorithm is evaluated against three related tracking methods on a set of seven challenging image sequences. Experimental results show that the LRSVT algorithm favorably performs against state-of-the-art tracking methods with regard to accuracy and execution time.

1 INTRODUCTION

Visual tracking finds a region in the current image that matches the given object. It is a well-known problem in computer vision with numerous applications including surveillance, driver assistance, robotics, human-computer interaction, and motion analysis (Zhang T et al. 2014). Despite demonstrated success, it remains challenging to design a robust visual tracking algorithm due to factors such as occlusion, background clutter, varying viewpoints, and illumination and scale changes (Wang L et al. 2015).

Recently, sparse and low-rank representation has cause for concern in many aspects (R. Xia et al. 2014, Zhang C et al. 2015). These tracking methods express a target by a sparse linear combination of the templates in a dictionary (Zhang T et al. 2014). These algorithms based on $\| \cdot \|_1$ minimization have been demonstrated time-consuming. Then they set up low-rank representation and sparse representation to solve the problem. However, they cannot balance the two characteristics in good reason. Sparse always pursue a sparse enough solution, which ignoring the low-rank structure. At the same time, $\| \cdot \|_{2,1}$ norm has been proved effective at represent both low-rank and sparse in some paper (Zhao M et al. 2014). Besides, the $\| \cdot \|_{2,1}$ norm avoid the time-consuming process of nuclear norm.

This paper, we use norm which can combine the low-rank and sparse characteristic to learn robust linear representations for efficient and effective object tracking. The proposed visual tracking algorithm is developed based on the particle filter. We can see the process in Fig. 1.

Fig. 1 shows the flowchart of the enforcement of the proposed algorithm by pruning particles. First, the target is selected from the first frame. Second, all particles are sampled based on the previous object. Third, the particles are pruned using the reconstruction error to prune particles. Finally, the object is selected using our LRSVT algorithm in the next frame, which enforces sparsely low-rank properties.
Object tracking is formulated as a sparse and low-rank representation problem from a new perspective, which is carried out by exploiting the relationship between the observations of the particle samples and jointly representing them using a dictionary of templates with an online update. The resulting sparsely low-rank representation of candidate particles facilitates robust performance for visual tracking. The relationship of these algorithms and the importance of each property for visual tracking are shown.

2 RELATED WORKS

The recent years have witnessed significant progress in tracking with sparse and low-rank representation. Most recently, an algorithm that jointly learns the sparse and low-rank representations of all particles (Zhang, K. et al. 2012; Zhang, T. et al. 2012–2014) is proposed for object tracking. Solutions to low-rank matrix minimization and completion problems have also achieved considerable progress. Zhou X et al (Zhou X et al. 2012) demonstrated that the image sequence of a cardiac cycle can be well approximated with a low-rank matrix. Zhang C (Zhang C et al. 2014) learned the observation model by extracting low-rank features. Yehui Yang et al (W Hu et al. 2016) developed a comprehensive study of the $L_2,1$ norm to tolerate the sudden changes between two adjacent frames that exploits the low-rank structure among consecutive target observations.

3 LOW RANK SPARSE VISUAL TRACKING

In this section, we present the proposed tracking algorithm based on low-rank sparse representations of particle samples.

3.1 Consistent Low-rank Sparse Representation

In this work, particles are sampled from previous object locations to predict the state $s_t$ of the target at time $t$, from which the region of interest $y_t$ is cropped in the current image and normalized to the template size. The state transition function $p(s_t | s_{t-1})$ is modeled by an affine motion model with a diagonal Gaussian distribution. The observation model $p(y_t | s_t)$ reflects a similarity between an observed image region $y_t$ corresponding to a particle $s_t$ and the templates of the current dictionary. In this paper, $p(y_t | s_t)$ is computed as a function of the difference between the consistent low-rank sparse representation of the target based on
object templates and its representation based on background templates. The particle that maximizes this function is selected as the tracked target at each time instance. At time $t$, $n_0$ sampled particles and corresponding vectorized gray-scale image observations form a matrix $X_0 = [x_1, x_2, ..., x_{n_0}]$, wherein the observation with regard to the $i$-th particle is denoted as $x_i \in \mathbb{R}^D$. Each observation is represented as a linear combination of templates from a dictionary $D_i = [d_1, d_2, ..., d_m]$, such that $X_0 = D_i Z_i$.

The columns of $Z_i = [z_1, z_2, ..., z_m]$ denote the representations of particle observations with regard to $D_i$. The dictionary columns contain templates used to represent each particle, including image observations of the tracked object and the background. Misalignment between dictionary templates and particle observations may lead to tracking drifts because representation is constructed on the pixel level. The dictionary $D_i$ can be constructed from an over-complete set using transformed templates of the target and background classes to alleviate this problem. This dictionary is also updated progressively. Temporal consistency is exploited to prune particles for efficient and effective tracking. A particle is considered temporally inconsistent if its observation is not linearly well represented by the dictionary $D_i$ and the representation of the tracked target in the previous frame, which is denoted as $Z_0$. More specifically, the particle is pruned in the current frame if the 2-norm of the reconstruction error $\|x_i - D_i Z_0\|$ is sufficiently large, thereby leaving a number of $x_i$; therefore, the number is set as $n$. In this work, temporal consistency is exploited as the appearances of the tracked object. Consequently, this process effectively reduces the number of particles to be represented from $n_0$ to $n$, where $n_0 \gg n$ in most cases. Next, the ones left after pruning are denoted as candidate particles, in which their corresponding observations are $X \in \mathbb{R}^{D \times n}$ and their representations are $Z \in \mathbb{R}^{m \times n}$.

The representation of each candidate particle is based on the following observations. (1) After pruning, the candidate particle observations can be modeled by a low-rank subspace (i.e., $X$ is low-rank); therefore, $Z$ (i.e., their representations with regard to $D_i$) is expected to be low-ranked. (2) The observation $x_i$ of a good candidate particle can be modeled by a small number of nonzero coefficients in its corresponding representation $Z_i$.

The aim of object tracking is to search for patches (with regard to particles) that have a representation similar to previous tracking results. Therefore, a “good” representation should be consistent over time. In the work of CLRST [4], the tracking problem is formulated by minimizing $Z$.

$$
\min_{Z,E} Z_i \|Z_i\|_1 + \lambda_1 \|Z_i - Z_{i-1}\|_1 + \lambda_2 \|E\|_1
$$

subject to $X = DZ + E$

where

$$
\|A\|_{p,q} = \left( \sum \left( \sum |A_{i,j}|^p \right)^{1/p} \right)^{1/q}
$$

In CLRST formulation, $\|Z\|_1$ is used to minimize the matrix rank of representations of all candidate particles together. Their sparse representation scheme
is $\|Z\|_1$, which has been shown to be robust to occlusion or noise in visual tracking. $\|Z\|_1$ is considered to replace $\lambda_1 \|Z\|_1 + \lambda_2 \|E\|_1$, which is the sparse congruency constraint on matrix $Z$. This constraint only allows a few rows of $Z$ to become nonzero, thereby deleting the ambiguous bases and maintaining principal bases. Therefore, the samples belonging to the same class are more likely to choose the same atom in their representation and share the same sparse pattern in their SR coefficient vectors. Thus, $Z$ is sparse and low-rank. By contrast, the sparse congruency constraint considers the global structure of $Z$ and eliminates rows of elements that have a slight contribution to the representation of the dataset and do not affect the low-rank structure of $Z$. Thus, the contribution time is greatly reduced (Zhao M et al. 2014).

### 3.1.2 Temporal $\|Z-Z_0\|_{2,1}$ and Reconstruction Error $\|E\|_{1,1}$

Temporal representation allows only a small number of particles to have representations different from the previous tracking results. The values and support of the columns in $E$ are informative because these values indicate the presence of occlusion (substantial values but sparse support) and determines whether a candidate particle is sampled from the background (substantial values with non-sparse support) (Zhao M et al. 2014).

### 3.2 Solving

#### 3.2.1 Solving Equation

$$L(Z_{1,2,3}, E, Y_{1,2,3}, u_{1,2,3}) = \lambda_2 \|Z\|_2 + \lambda_3 \|Z\|_{2,1} + \lambda_4 \|E\|_1 + \text{tr}[Y^T_1 (X - DZ_1 - E)] + \frac{\mu_2}{2} \|X - DZ_1 - E\|_F + \text{tr}[Y^T_2 (Z_1 - Z_1)] + \frac{\mu_3}{2} \|Z_1 - Z\|_F + \text{tr}[Y^T_3 (Z_2 - Z_2 - Z_0)] + \frac{\mu_4}{2} \|Z_2 - Z_2 - Z_0\|_F$$

(5)

#### 3.2.2 Solving $Z_1, Z_2, E, Z_3$ in Turn

$$Z_1^* = \arg\min_{u_2} \frac{\lambda_2}{u_2} \|Z_1\|_2 + \frac{1}{2} \|Z_1 - Z_2 - \frac{1}{u_2} Y_2\|_F$$

(6)

$$Z_1^* = \mathcal{L}_{\frac{\lambda_2}{u_2}} (Z_1 + \frac{1}{u_2} Y_2)$$

(7)

$$Z_2^* = \arg\min_{u_3} \frac{\lambda_3}{u_3} \|Z_2\|_{2,1} + \frac{1}{2} \|Z_2 - Z_1 - \frac{1}{u_3} Y_3\|_F$$

(8)

$$Z_2^* = \mathcal{L}_{\frac{\lambda_3}{u_3}} (Z_2 - Z_1 - \frac{1}{u_3} Y_3)$$

$$E^* = \arg\min_{u_1} \frac{\lambda_4}{u_1} \|E\|_1 + \frac{1}{2} \|E - X + DZ_1 - \frac{1}{u_1} Y_1\|_F$$

(9)

$$E^* = \mathcal{L}_{\frac{\lambda_4}{u_1}} (E - X + DZ_1 - \frac{1}{u_1} Y_1)$$

And

$$Z_3^* = \arg\min_{u_1} \|Y^T (X - DZ_1 - E)\|_F + \frac{\mu_2}{2} \|X - DZ_1 - E\|_F + \text{tr}[Y^T (Z_1 - Z_1)] + \frac{\mu_3}{2} \|Z_1 - Z_1\|_F + \text{tr}[Y^T (Z_2 - Z_2 - Z_0)] + \frac{\mu_4}{2} \|Z_2 - Z_2 - Z_0\|_F$$

$$Z_3^* = \mathcal{L}_{\frac{\lambda_1}{u_1}} (Z_3 - Z_1) + \frac{1}{u_1} Y_1$$

#### 3.2.3 Update $Y_{1,2,3}, u_{1,2,3}$

$$\begin{aligned}
Y_1 &= Y_1 + u_1 (X - DZ_1 - E) \\
Y_2 &= Y_2 + u_2 (Z_2 - Z_1) \\
Y_3 &= Y_3 + u_3 (Z_3 - Z_2 - Z_0) \\
u_1 &= \rho u_1; u_2 = \rho u_2; u_3 = \rho u_3;
\end{aligned}$$

#### 3.3 Adaptive Dictionary

The dictionary $D_j$ is initialized by sampling image patches around the initial target position.
dictionary is updated in successive frames to model the change in appearance of the target object and to ensure accuracy in the tracking. $D_t$ is augmented with representative templates of the background to alleviate the problem of tracking drift, such that $D_t = [D_O, D_B]$, where $D_O$ and $D_B$ represent the target object and background templates, respectively. Thus, the representation $z_k$ of a particle comprised an object representation $z_k^O$ and a background representation $z_k^B$. The tracking result $y_i$ at instance $t$ is the particle $x_i$, such that

$$i = \arg \max_{k=1,...,n} \left( \|z_k^O\|_2 - \|z_k^B\|_2 \right)$$ (14)

which encourages good modeling of the tracking result using object templates and not using background templates. Discriminative information was also employed to design a systematic procedure for updating $D_t$.

## 4 EXPERIMENT

In this section, the experimental results on the evaluation of the proposed tracking algorithm against several state-of-the-art methods were evaluated.

### 4.1 Datasets

Twenty-five challenging videos with ground truth object locations, including basketball, football, singer1, singer2, singer1 (low frame rate), skating1, and skating2 were used for analysis. These videos contain complex scenes with challenging factors (e.g., cluttered background, moving camera, fast movement, large variation in pose and scale, occlusion, shape deformation, and distortion).

### 4.2 Evaluated Algorithms

The proposed tracking methods (SLRVT) are compared with three state-of-the-art visual trackers, including FCT (Zhang K et al. 2014), $I_t$ (Zhao M et al. 2014), and CLRST (Mei X et al. 2011). Publicly available sources or binary codes provided by the authors are used for fair comparisons. The same initialization and parameter settings in all experiments are also used.

### 4.3 Evaluation Criteria

Two metrics are used to evaluate tracking performance. The first metric is the center location error, which is the Euclidean distance between the central location of a tracked target and the manually labeled ground truth. The second metric is an overlap ratio based on the PASCAL challenge object detection score (Everingham B M. et al. 2010). Given tracked bounding box $ROI_t$ and the ground truth bounding box $ROI_{GT}$, the overlap score can be computed as

$$score = \frac{\text{area}(ROI_t \cap ROI_{GT})}{\text{area}(ROI_t \cup ROI_{GT})}$$ (15)

The average overlap score across all frames of each image sequence is computed to rank the tracking performance.

### 4.4 Implementation Details

All experiments are carried out in MATLAB on a 3.2 GHz Intel Corei5–4460 Duo machine with 4 GB RAM. Template size $d$, which is manually initialized in the first frame, is set to half the size of the target object. The affine transformation, where the state transitional probability $p(y_i | s_{i-1})$ is modeled by a zero-mean Gaussian distribution and a diagonal covariance matrix $\sigma$ with values $(0.03, 0.0005, 0.0005, 0.03, 1, 1)$: $p(y_i | s_{i-1}) \sim \mathcal{N}(0, \sigma)$, is used.

The definition of $p(y_i | s_i)$ is

$$p(y_i | s_i) = \delta z_i (i = 1, 2, ..., n)$$

The representation threshold is set to 0.5. Parameter $\sigma$ is set to 1.0 in the CLRST method to prune particles. The number of particles $n_0$ is set to 400 and total number of templates $m$ is set to 25.

## 5 TEST RESULTS

### 5.1 Parameter Analysis

Several parameters play important roles in the proposed tracking algorithm. In this section, determining the values and effects of these parameters on tracking performance is shown.

Effect of $\lambda$: The objective function has three parameters, namely,
Table 1: a. The distance value with the change of $\lambda_1$. b. The distance value with the change of $\lambda_3$.

<table>
<thead>
<tr>
<th>$\lambda_1$:</th>
<th>0.0001</th>
<th>0.5</th>
<th>0.9</th>
<th>1</th>
<th>1.1</th>
<th>2</th>
<th>5</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance</td>
<td>50.1</td>
<td>54</td>
<td>50.6</td>
<td>33.7</td>
<td>54.3</td>
<td>53.2</td>
<td>55.8</td>
<td>58.4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\lambda_3$:</th>
<th>0.0001</th>
<th>0.5</th>
<th>0.9</th>
<th>1</th>
<th>1.1</th>
<th>2</th>
<th>5</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance</td>
<td>55.5</td>
<td>54</td>
<td>51.5</td>
<td>41.7</td>
<td>54.6</td>
<td>44.8</td>
<td>52</td>
<td>51.1</td>
</tr>
</tbody>
</table>

Figure 2: Tracking result on 7 image sequences. LRSVT, FCT (Zhang K et al. 2014), CLRST (Zhao M et al. 2014), $I_1$ (Mei X et al. 2011) are respectively displayed in red, green, blue and yellow.
Table 2: Overlap, distance and time result on 7 image sequences with LRSVT, CLRST, FCT and $l_1$ methods.

<table>
<thead>
<tr>
<th>a. Overlap</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LRSVT</td>
<td>CLRST</td>
<td>$l_1$</td>
<td>FCT</td>
</tr>
<tr>
<td>football(280fps)</td>
<td>0.262</td>
<td>0.312</td>
<td>0.582</td>
<td>0.283</td>
</tr>
<tr>
<td>basketball(141fps)</td>
<td>0.381</td>
<td>0.506</td>
<td>0.661</td>
<td>0.501</td>
</tr>
<tr>
<td>singer2(366fps)</td>
<td>0.436</td>
<td>0.588</td>
<td>0.408</td>
<td>0.328</td>
</tr>
<tr>
<td>singer1(351fps)</td>
<td>0.436</td>
<td>0.588</td>
<td>0.818</td>
<td>0.047</td>
</tr>
<tr>
<td>skating2(707fps)</td>
<td>0.284</td>
<td>0.17</td>
<td>0.101</td>
<td>0.255</td>
</tr>
<tr>
<td>singer1(low frame rate)(181fps)</td>
<td>0.592</td>
<td>0.095</td>
<td>0.344</td>
<td>0.255</td>
</tr>
<tr>
<td>skating1(285fps)</td>
<td>0.511</td>
<td>0.642</td>
<td>0.411</td>
<td>0.669</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>b. Distance</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LRSVT</td>
<td>CLRST</td>
<td>$l_1$</td>
<td>FCT</td>
</tr>
<tr>
<td>football(280fps)</td>
<td>5.2</td>
<td>4.2</td>
<td>11.4</td>
<td>6</td>
</tr>
<tr>
<td>basketball(141fps)</td>
<td>9</td>
<td>23.3</td>
<td>23</td>
<td>16.5</td>
</tr>
<tr>
<td>singer2(366fps)</td>
<td>130</td>
<td>133.7</td>
<td>35.1</td>
<td>49.7</td>
</tr>
<tr>
<td>singer1(351fps)</td>
<td>130</td>
<td>133.7</td>
<td>3.5</td>
<td>22.4</td>
</tr>
<tr>
<td>skating2(707fps)</td>
<td>98.4</td>
<td>64.2</td>
<td>184.8</td>
<td>127.6</td>
</tr>
<tr>
<td>singer1(low frame rate)(181fps)</td>
<td>4.7</td>
<td>162.7</td>
<td>37</td>
<td>127.6</td>
</tr>
<tr>
<td>skating1(285fps)</td>
<td>24.4</td>
<td>8.3</td>
<td>35.9</td>
<td>18.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>c. Time</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LRSVT</td>
<td>CLRST</td>
<td>$l_1$</td>
<td>FCT</td>
</tr>
<tr>
<td>football(280fps)</td>
<td>0.431</td>
<td>1.597</td>
<td>0.00027</td>
<td>0.0178</td>
</tr>
<tr>
<td>basketball(141fps)</td>
<td>0.563</td>
<td>2.354</td>
<td>0.000269</td>
<td>0.0195</td>
</tr>
<tr>
<td>singer2(366fps)</td>
<td>0.59</td>
<td>2.555</td>
<td>0.000089</td>
<td>0.0174</td>
</tr>
<tr>
<td>singer1(351fps)</td>
<td>0.59</td>
<td>2.555</td>
<td>0.000136</td>
<td>0.0186</td>
</tr>
<tr>
<td>skating2(707fps)</td>
<td>0.821</td>
<td>1.608</td>
<td>0.000085</td>
<td>0.019</td>
</tr>
<tr>
<td>singer1(low frame rate)(181fps)</td>
<td>0.581</td>
<td>2.65</td>
<td>0.000194</td>
<td>0.019</td>
</tr>
<tr>
<td>skating1(285fps)</td>
<td>0.586</td>
<td>2.674</td>
<td>0.000216</td>
<td>0.018</td>
</tr>
</tbody>
</table>

$\lambda_1$, $\lambda_3$, and $\lambda_4$ (1). Because $\lambda_1$ and $\lambda_3$ are related to the coefficients $Z$ and $\lambda_4$ is related to $E$, $\lambda_4 = 1$ was fixed and other parameter values were changed. All $\lambda_i$ ($i=1,3$) are parameterized by a discrete set $\Lambda$ for sensitivity analysis, in which $\Lambda = \{0.0001, 0.5, 0.9,1, 1.1, 2, 5, 10.0\}$. The different combinations of these values were analyzed on video with 100 frames. The average distance score from all frames was computed for each combination. The corresponding results were obtained for different $\lambda_i$, as shown in Table 1.a. Table 1 shows the sensitivity analysis of $\lambda_i$ ($i=1,3$). From these results, we can set $\lambda_i = 1$, $\lambda_3 = 1$, and $\lambda_4 = 1$ for the objective function (1).

5.2 Qualitative Comparison

Fig. 2 and Table 2 show the tracking results of four trackers on seven sequences. Three norms are included: overlap, distance, and time. Singer1(low frame rate) has better tracking performance based on the visual effect of the views of football, basketball, and singer1. The proposed method performed well in terms of position and size of the target. The singer2 sequence contains significant illumination, scale, and viewpoint changes. skating2 contains Abrupt Motion, Illumination Change, and Occlusion. Therefore, most trackers drift away from the target object in these two sequences. In the Singer2 sequence, only the result of the LRSVT method falls on the screen. In Skating1 sequence, length and width did not fully track the target in terms of the basic location of the tracking target.
LRSVT performed well at overlap in singer1 (low frame rate) and at the distance in basketball than any of the other methods. Among all sequences, the time consumed from fastest to slowest is in the order of $I_1$, FCT, LRSVT, and CLRST.

6 CONCLUSION

This paper conducted based on the CLRST method. $I_{2,1}$ norm was used to represent low-rank and sparse, which differs from CLRST. The performance of the tracking algorithms against three competing state-of-the-art methods on seven challenging image sequences was analyzed extensively. The proposed method significantly reduced computation time than CLRST. The result maintained more than twice the speed of operation with the same overlap and distance. The results are in line with expectations.

ACKNOWLEDGEMENT

This research was partially sponsored by National Natural Science Foundation (NSFC) of China under project No.61403403 and No.61402491.

REFERENCES


Zhang, T., Ghanem, B., & Ahuja, N. (2012). Robust multiobject tracking via cross-domain contextual information for sports video analysis. 22(10), 985-988.


