“© 2014 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

VISUAL TRACKING VIA GRAPH-BASED EFFICIENT MANIFOLD RANKING WITH LOW-DIMENSIONAL COMPRESSIVE FEATURES

Tao Zhou[1], *Xiangjian He*[2], *Kai Xie*[1], *Keren Fu*[1], *Junhao Zhang*[1], *Jie Yang*[*1]

[1]Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, China
[2]Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia

ABSTRACT

In this paper, a novel and robust tracking method based on efficient manifold ranking is proposed. For tracking, tracked results are taken as labeled nodes while candidate samples are taken as unlabeled nodes, and the goal of tracking is to search the unlabeled sample that is the most relevant with existing labeled nodes by manifold ranking algorithm. Meanwhile, we adopt non-adaptive random projections to preserve the structure of original image space, and a very sparse measurement matrix is used to efficiently extract low-dimensional compressive features for object representation. Furthermore, spatial context is used to improve the robustness to appearance variations. Experimental results on some challenging video sequences show the proposed algorithm outperforms six state-of-the-art methods in terms of accuracy and robustness.

Index Terms— visual tracking, appearance model, manifold ranking, random projections, low-dimensional compressive features, spatial context

1. INTRODUCTION

Visual tracking is a long standing research topics due to its wide range of applications such as behavior analysis, activity recognition, video surveillance, and human-computer interaction [1]. Although significant progress has been obtained in the past decades, developing an efficient and robust tracking algorithm is still a challenging problem due to numerous factors such as partial occlusion, illumination variation, pose change, abrupt motion, and background clutter.

The main tracking algorithms can be classified into two kinds: generative [2, 3, 4, 5] or discriminative methods [6, 7, 8, 9]. Generative methods focus on searching for the regions which are the most similar to the tracked targets. While it is critical to construct an effective appearance model in order to handle various challenging factors in tracking, the involved computational complexity is often increased at the same time. Furthermore, generative methods discard useful information surrounding target regions that can be exploited to better separate objects from backgrounds. Discriminative methods cast tracking as a classification problem that distinguishes the tracked targets from the surrounding backgrounds. Above tracking methods have shown promising performance. However, their main shortcomings are as follows: Firstly, the effective searching algorithm and measured method between object model and candidate samples are difficult to obtain in generative method. Secondly, the aim of discriminative methods is to distinguish the target region from complicated background, but background varies broadly during the tracking process or there exists similarity between object and background. Thus, it is very difficult to construct a discriminative object representation. Thirdly, feature selection is of crucial importance for generating an effective appearance model, but current many features make the computational load very heavy.

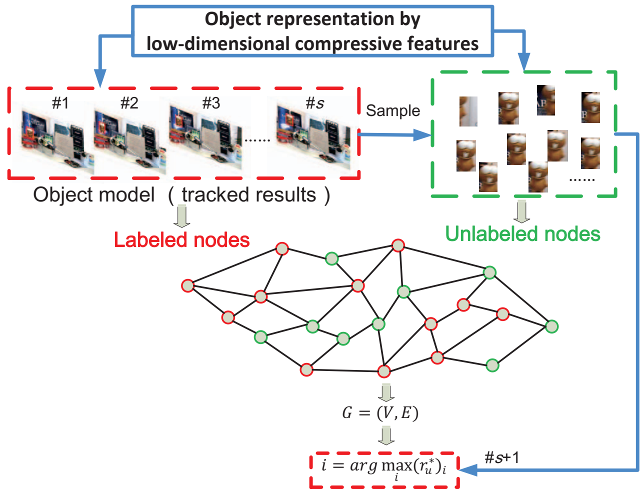

Object representation by
low-dimensional compressive features
#1
#2
#3
#s
......
Sample
Object model (tracked results)
Labeled nodes
Unlabeled nodes
$G = (V, E)$
$i = arg \max_i (r_u^*)_i$
#s+1

Fig. 1. Basic flow of our tracking algorithm. A graph is established combined labeled nodes (tracked results) and unlabeled nodes (candidate samples), and ranking scores represent the relevance between object model and candidate samples.

Motivated by the success of a graph-based ranking algorithm, it has been widely applied in information retrieval and shown to have excellent performance and feasibility on a variety of data types [10, 11, 12]. Manifold ranking algorithm firstly constructs a weighted graph by using each data node as

[*]Corresponding author: Jie Yang, jieyang@sjtu.edu.cn. This research is partly supported by NSFC, China (No:61273258, 61105001).

a vertex. The ranking score of the query is iteratively propagated to nearby node via the weighted graph. Finally nodes will be ranked according to the ranking scores, in which a larger score indicates higher relevance. In this paper, we develop a novel and robust tracking method based on manifold ranking, which regards tracking as a ranking problem. As shown in Fig.1, we note the tracked results as labeled nodes, while candidate samples are regarded as unlabeled nodes. The tracking objective is to estimate corresponding likelihood that is determined by the relevance between the queries and all candidate samples. Our method is different with [13] in that, we use manifold structure to measure relevance between model and samples and low-dimensional compressive features can efficiently compress features from the foreground objects and background ones. Experimental results on some challenging video sequences with comparisons to state-of the-art tracking six methods demonstrate the effectiveness and robustness of the proposed model and algorithm.

The main contributions of this paper are as follows:(1) A novel graph-manifold ranking based visual tracking method is proposed in this paper. (2) Efficient manifold ranking algorithm is adopted in our proposed method, it can reconstruct graph efficiently in each tracking round and reduce the computation complexity. (3) Low-dimensional compressive features are extracted by a very sparse measurement matrix for object representation, which preserve the structure of original image space and discriminate object from clutter background effectively. (4) Our method exploits temporal and spatial context information, which is robust to appearance variations introduced by abrupt motion, occlusion,and pose variations.

2. GRAPH-BASED MANIFOLD RANKING

The manifold ranking method is described as follows: given a query node, the remaining unlabeled nodes are ranked based on their relevance to the given query. The goal is to learn a ranking function to define the relevance between unlabeled nodes and this query [11, 12]. In [12], a ranking method that exploits the intrinsic manifold structure of data for graph labelling is proposed. Given a data set $X = \{x_1, x_2, \cdots, x_l + 1, \cdots, x_n\} \in \Re^{m \times n}$, some data points are labelled queries and the rest need to be ranked according to their relevance to the queries. $W \in \Re^{n*m}$ denotes the adjacency matrix with element W_{ij} that indicates the weight of the edge between point i and j. Generally, the weight can be defined by the kernel $w_{ij} = e^{-d^2(x_i, y_j)/2\sigma^2}$ if there is an edge linking x_i and y_j, otherwise $w_{ij} = 0$. The function $d(x_i, y_j)$ represents a distance metric between x_i and y_j.

Let $f : X \to \Re^n$ denotes a ranking function which assigns a ranking value r_i to each point x_i , and r can be defined as a vector $r = [r_1, r_2, \cdots, r_n]^T$. Let $y = [y_1, y_2, \cdots, y_n]^T$ denote an indication vector, in which $y_i = 1$ if x_i is a query, and $y_i = 0$ otherwise. Suppose all data points represent a graph $G = (V, E)$, where V represents vertex set, and E represents the edge set with $W = W_(ij), i, j = 1, 2, \cdots, N$. The strength of edge reflects the similarity between two vertices. To solve the optimal ranking of queries, the cost function associated with f is defined as follows:

$$O(r) = \frac{1}{2}(\sum_{i,j=1}^{n} \| \frac{1}{\sqrt{D_{ii}}} r_i - \frac{1}{\sqrt{D_{jj}}} r_j \|^2 + \mu \sum_{i=1}^{n} \|r_i - r_j\|^2) \quad (1)$$

where $\mu > 0$ is the regularization parameter and D is a diagonal matrix with the element $D_{ii} = \sum_{j=1}^{n} w_{ij}$. To minimize the cost function, we can obtain the closed form solution as:

$$r^* = (I - \alpha S)^{-1} y \quad (2)$$

where I is an identity matrix, $\alpha = \frac{1}{1+\mu}$ and $S = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$. Then, we use the iteration scheme to solve this optimal problem:

$$r(t+1) = \alpha S(r(t) + (1 - \alpha) y) \quad (3)$$

where α is control parameter, which balances each points information from its neighbors and that of initial information.

3. OUR PROPOSED METHOD

3.1. Framework

Fig.1 shows the basic flow of our proposed tracking algorithm. The tracking problem is formulated as a ranking task. Firstly, we assume the location in the first t frames have been obtained by CT tracker [6]. Let $l(x_i^*)$ denote the location of tracking result at the i-th frame where x_i^* represents the sample. Then we collect these tracked results as object appearance model set $S_m = \{x_1^*, x_2^*, \cdots, x_i^*\}, i = 1, 2, \cdots, t$, and the corresponding graph is taken as G_m. Secondly, for a new frame, we crop out a set of image patches x^r with N samples near the location $l(x_t^*)$ with a search radius at current frame, i.e.$x^\beta = \{x : \|l(x) - l_t(x^*))\| < \beta\}$. These candidate image patches are collected as unlabeled nodes $S_u = \{x_1^{s+1}, x_2^{s+1}, \cdots, x_i^{s+1}\}, i = 1, 2, \cdots, N$, and the corresponding graph is taken as G_u. Thirdly, the candidate G_u is combined with G_m to construct a graph $G = G_m \cup G_u$, in which the label $y_i = 1$ if a node point is in G_m, and $y_i = 0$ if a node point is in G_u. The ranking score $r^* = [r_m^*; r_u^*]$ can be obtained by manifold ranking algorithm, where r_m^* is corresponding to G_m while r_u^* is corresponding to G_u. Then, the tracking result is added into S_m, while the other candidate samples are deleted. This procedure continues to sample candidates and construct a new graph to obtain the largest ranking score as tracking result until the end of the image sequence.

3.2. low-dimensional compressive features

The Haar-like features have been widely used for object detection and object tracking [8, 6, 14]. However, Haar-like features require high computational loads for feature extraction in training and tracking phases. Recently, Babenko et

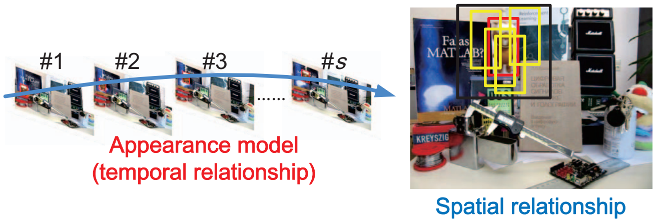


Fig. 2. Temporal and spatial relationship.

al. [8] adopted the generalized Haar-like features where each one is a linear combination of randomly generated rectangle features, and use online boosting to select a small set of them for object tracking. In our tracking framework, we use the low-dimensional compressive features proposed by Zhang et al. for the appearance model [6].

3.3. Appearance model updating process

As shown in Fig.1, we can obtain the locations in the first t frames by CT tracker, and then to obtain the location of the $t+1$ frame by manifold ranking algorithm. There exists an obvious problem that the size of S_m will be very large if all tracked results are added into appearance model in each tracking round, so the computation complexity will be very heavy. In addition, the bad node impacts the performance of appearance model. To track the next frame, we need to update appearance model firstly. We compute the average ranking score of r_m^*;

$$\mu_{r_m^*} = \sum_{i=1}^{t} (r_m^*)_i \tag{4}$$

where $(r_m^*)_i$ represents the score of the r-th node in S_m. Then to compute displacement error e_i between the score of S_m and the average score:

$$e_i = \|(r_m^*)_i - \mu_{r_m^*}\|^2 \tag{5}$$

We delete the node that has the largest displacement error, and then add current tracking result x_{t+1}^* into S_m. Thus, the number of S_m will be t constantly. It is worth noting that the average ranking score computed from tracked results alleviates the noise effects.

3.4. Support set construction

In our method, object appearance model S_m only reflects the temporal relationship among consecutive frames, while it can not consider its immediate surrounding background. In tracking process, the context of a target in an image sequence consists of the spatial context including the local background and the temporal context including all appearances of the targets in previous frames. As shown in Fig.2 (left), our object appearance model S_m represents the temporal context by previous frames. In Fig.2 (right), note that the object can be influenced by its surrounding background, there exists the correlation between the object (denoted by red rectangle) and its surrounding background (denoted by yellow rectangle). Therefore, in order to make use of surrounding background information and provide much appearance information for construct graph, we establish a support set to describe the spatial context. The spatial context describes the relevance the object and its surrounding background in small neighborhood region.

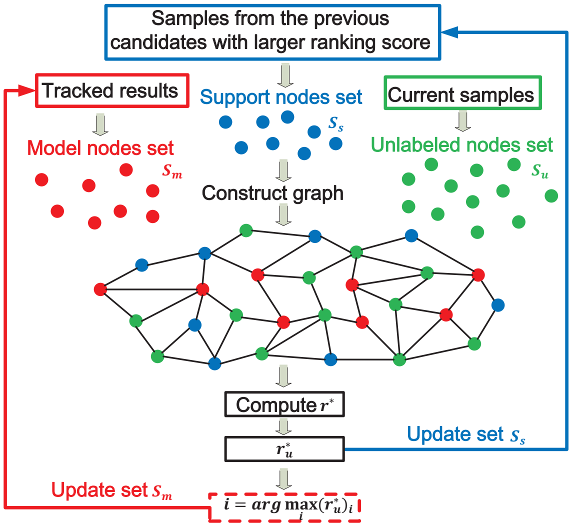


Fig. 3. Appearance model updating process and support set construction.

Supposed in tracking the $t+1$ frame, we have obtained the object location $l(x_{t+1}^*)$ by ranking score, and the ranking score of the current candidate samples is denoted as r_u^*. We select s nodes from candidate samples set S_u to construct the support set S_s. S_s are corresponding to the first $s+1$ largest ranking score in r_u^*, and then we delete the largest one. The graph corresponding to support set is denoted as G_s. The appearance model updating process and support set construction are shown in details in Fig.3.

To track the $t+2$ frame, a graph $G = G_m \cup G_s \cup G_u$ is constructed and the label $y_i = 1$ if a node point is from S_m and S_s, while $y_i = 0$ if a node point is from S_u. The ranking score $r^* = [r_m^*; r_s^*; r_u^*]$ can be obtained by efficient manifold ranking algorithm (see in section 4), where r_m^*, r_s^*, and r_u^* are corresponding to G_m, G_s, G_u respectively. The tracking scheme is summarized in Algorithm 1. Finally, the target in frame $t+2$ is the sample with the largest component in r_u^*, as the i-th sample can be selected from S_u computed by:

$$i = \operatorname*{argmax}_i r_u^*, i = 1, 2, \cdots, N \tag{6}$$

Algorithm 1. The proposed tracking method

```
Input: Video frame $f$=1:$F$
The first $t$ frames are tracking by CT tracker to
  construct object appearance model set
  $S_m = \{x_1^*, x_2^*, \cdots, x_i^*\}$
for $f$= $t$+1 to $F$ do
  Crop out a set of candidate samples as unlabeled set
    $S_u$ by $x^\beta = \{x : \|l(x) - l_t(x^*))\| < \beta\}$.
  if $f$ == $t$+1
    Construct a graph $G = G_m \cup G_u$ and support set $S_s$.
    Update model set $S_m$.
  else
    Construct a graph $G = G_m \cup G_s \cup G_u$.
    Update model set $S_m$ and support set $S_s$.
  end if
  The $i$-th candidate sample that has the largest in $r_u$ is
    taken as the object location, as the largest score is
    defined as $i = \mathrm{argmax}_i\, r_u^*, i = 1, 2, \cdots, N$.
end for
Output: Tracking results $\{l_1(x^*), l_2(x^*), \cdots, l_F(x^*)\}$.
```

4. EFFICIENT MANIFOLD RANKING ALGORITHM

In order to efficiently reconstruct graph, we use efficient manifold ranking algorithm [12] to compute the ranking score. First, we briefly introduce how to use anchor graph to model the data. Given a data set $X = \{x_1, x_2, \cdots, x_n\} \in \Re^{m \times n}$, $U = \{u_1, u_2, \cdots, u_d\} \in \Re^{m \times d}$ indicates a set of anchors sharing the same space with the data set. Then, we define a real value function $r : X \to R$, which assigns a semantic label for each point in X. The aim is to find a weight matrix that measures relevance between data points $r : X \to R$ and anchors in U. We obtain $r(x)$ for each point by a weighted average of these labels on anchors as follows:

$$r(x_i) = \sum_{k=1}^{d} z_{ki} r(u_k), i = 1, 2, \cdots, n \tag{7}$$

where $\sum_{k=1}^{d} z_{ki} = 1$ and $z_{ki} > 0$, in which z_{ki} represents the weight between point x_i and an anchor u_k. The weights can be obtained by Nadaraya-Watson kernel regression to increase smoothness. The graph construction process and the means how to get the anchors can be found in detailed [12].

The weight matrix $Z \in \Re^{d*n}$ can be viewed as a d-dimensional representation of the data $X \in \Re^{m*n}$, in which d is the number of anchor points. It means that data points can be presented in a new space to replace the original feature space. We set the adjacency matrix as follows:

$$W = Z^T Z \tag{8}$$

where $W_{ij} > 0$ if two points are correlative and they will share at least one common anchor point, otherwise $W_{ij} = 0$. The new adjacency matrix is useful to explore relevance among data points. According to $W = Z^T Z$, the equation 2 can be rewritten as follows:

$$r^* = (T_1 - \alpha H^T H)^{-1} y = (I_1 - H^T(HH^T - \frac{1}{\alpha} I_2))y \tag{9}$$

with

$$D_{ii} = \sum_{j=1}^{n} w_{ij} = \sum_{j=1}^{n} z_i^T z_j = z_i^T v \tag{10}$$

where $H = ZD^{-1}$, and $S = H^T H$. I_1 and I_2 are the identity matrices. From the above equations, we obtain the matrix D without using the matrix W. Due to a low complexity for computing the ranking function r^*, we can reconstruct graph in each tracking round efficiently.

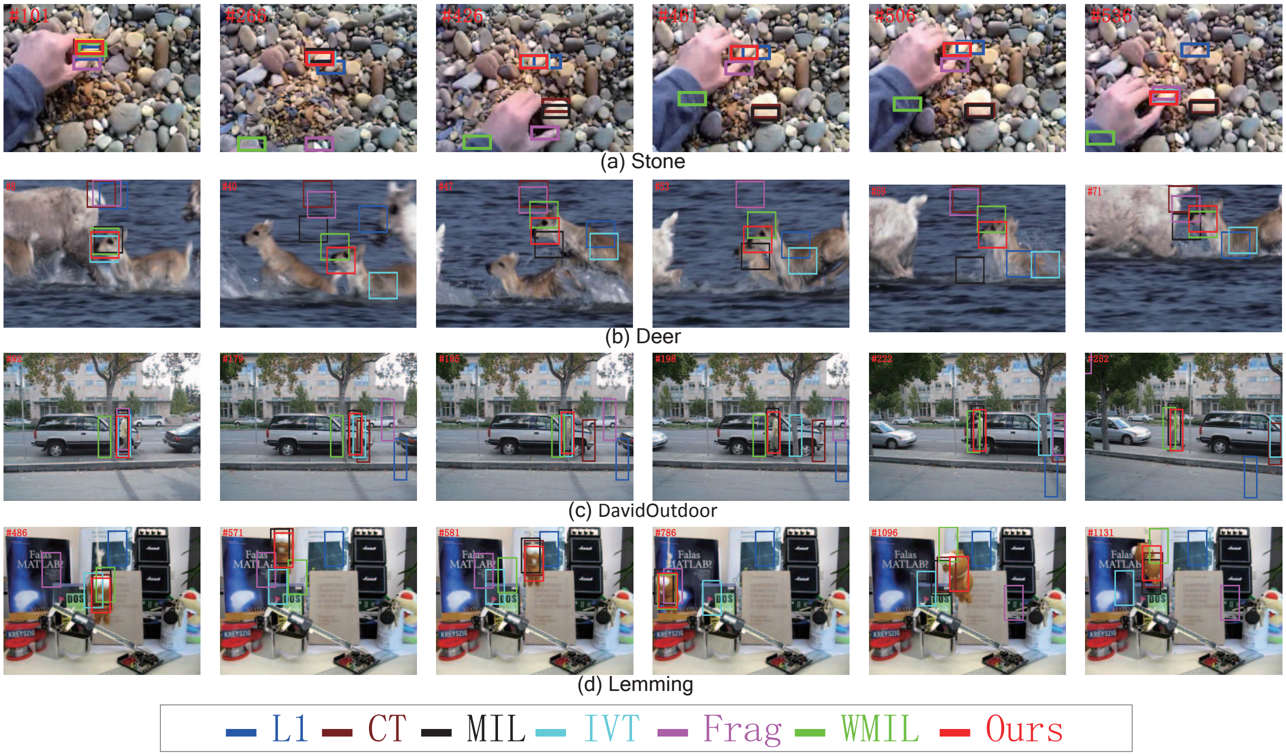
5. EXPERIMENTAL RESULTS AND ANALYSIS

5.1. Experimental setup

We evaluate the proposed tracking method based on efficient manifold ranking algorithm and object representation with low-dimensional features using four video sequences with impacted factors including abrupt motion, cluttered background, appearance and pose variation. We compare our proposed tracker with six other state-of-the-art methods including: L1 tracker (L1) [4], real-time compressive tracking (CT) [6], multiple instance learning tracker (MIL) [3], incremental visual tracking (IVT) [3], fragment tracker (Frag) [2], and weighted multiple instance learning tracker (WMIL) [9]. For fair comparison, we adopt the source or binary codes provided by the authors with tuned parameters for best performance. But for some trackers involving randomness, we repeat the experimental results 5 times on each sequence and obtain the averaged results. In our experiments, the parameters are used in our algorithm as follows: the search radius for cropping out candidate samples is set to $\beta = 20$, which is related with object motion speed. The dimensionality of compressive feature is set 100. The first t frames are tracking by CT method and t is set to 30, and the number of nodes in support set is set $s = 10$. Implemented in MATLAB, our tracking method runs at 12 frames per second (FPS) by averaged results on an i3 3.20 GHz machine with 4 GB RAM.

5.2. Experimental results

Table.1 reports the center location error, where smaller CLE means more accurate tracking results. Table.1 shows the quantitative results in which our tracking algorithm achieves the better performance. Fig.4 shows part of tracking results by different tracking methods and more results can be found in the supplementary materials. Fig.5 illustrates the tracking results in terms of center location error, which is defined as the Euclidian distance between the center location obtained by tracking algorithm and the ground truth. Overall, our method performs favorably against the other state-of-the-art tracking methods.



(a) Stone

(b) Deer

(c) DavidOutdoor

(d) Lemming

Fig. 4. Screenshots of some sampled tracking results.

Table 1. Center location error (CLE) (in pixels). Red fonts indicate the best performance while the blue fonts indicate the second best ones.

Sequence	L1	CT	MIL	IVT	Frag	WMIL	Ours
DavidOutdoor	100.4	87.3	38.4	52.9	90.5	73.3	29.5
Deer	171.5	95.1	66.5	127.5	92.1	25.1	23.0
Lemming	184.9	26.3	25.9	93.4	149.1	96.9	24.3
Stone	19.2	32.8	32.3	2.5	65.9	99.8	6.4
Average CLE	119.0	60.4	40.8	69.1	99.4	73.8	20.8

Abrupt Motion: The object in Deer and Lemming sequences has an abrupt motion. Only our method performs well on Deer sequence, while other trackers suffer severe drift at frames #40, #47, #53, #59, #71. In Lemming sequence, only MIL, CT and our method perform well at frame #486, while the other algorithms fail to track the target objects well. Similarly, other some trackers fail to track the target objects well but our method can still obtain tracking performance as at frames #1096 and #1131.

Background Clutter: The trackers are easily confused if the object is very similar to the background. Fig.4 (a), (b), and (d) demonstrate the tracking results in the Stone, Deer and Lemming sequences with background clutter. Fig.4 (a) shows different trackers tracking a yellow cobblestone located among a lot of similar stones. Thus, it is very difficult to distinguish object from background and keep track of the objects correctly. Comparatively, our method exhibits better discriminative ability and outperforms other at frames #426, #461, #506, #536 in Stone sequence, while the some other algorithms fail to track the target objects well. The plot of position error is presented in Fig.4 (Stone), which demonstrates that the result of our method is very close to the ground truth. The MIL and WMIL tracker completely drifts to the background at frames #266, #426, #461, #506, #536, which verifies that the selected features by the MIL tracker are less informative than our method. The Frag tracker has severe drift at frames #426, #461 and #506 because its template does not update online, making it unable to handle large background clutter. The CT has severe drift at frames #461, #506 and #536 because it only uses compressive feature and Bayesian classifier is sensitive to background clutter. In Deer sequence, our method outperforms all other methods in the whole given frames.

Partial Occlusion: Fig.4 (c) demonstrates that the proposed method performs well in terms of position and rotation when the target undergoes partial occlusion. Our method and MIL perform better than other methods at frame #198, #222 and #252, while other methods suffer from sever drift and some of these methods completely fail to track. But our method performs more accurate than MIL at frame #198 and #252. Thus, our method can handle occlusion and it is not sensitive to partial occlusion since manifold ranking algorithm can measure the relevance between object appearance and candidate samples.

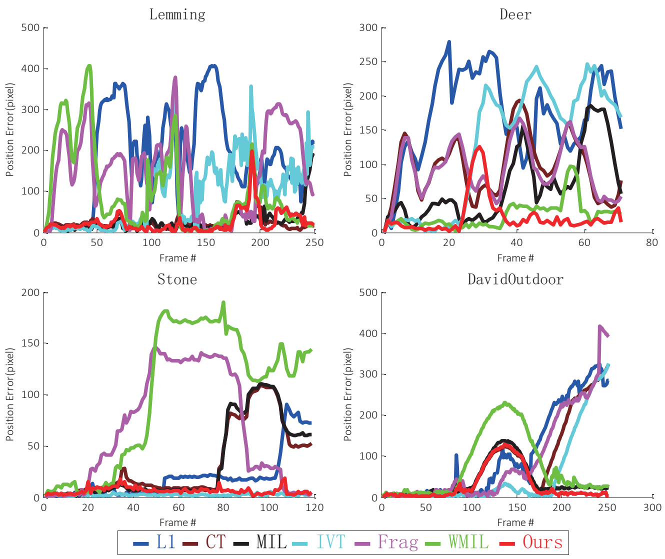


Fig. 5. Error plots for video sequences.

5.3. Discussion

As shown in our experiments, our method can address these factors including abrupt motion, cluttered background, partial occlusion more effectively. The reasons are as follows: (1) Discriminative features are extracted by a very sparse matrix to separate object well from background sample, and the low-dimensional compressive features can preserve the structure of original image space. (2) The outstanding ability of manifold ranking algorithm is to discover underlying geometrical structure and relevance between object appearance and candidate samples. (3) Our method combines temporal with spatial context information for tracking, it is very insensitive to multiple factors.

However, the ranking score can not indicate the relevance under similar appearance information between object and non-object. Therefore, our method can not distinguish object from background clutters with some tracking error or failures as #266 and #536 in Fig.4(a).

6. CONCLUDING REMARKS

This paper proposed a novel framework named manifold ranking based visual tracking. In order to address the shortcomings of original manifold ranking from graph reconstruction and heavy computation load, we adopt the efficient manifold ranking algorithm. The ability of efficiently constructing a graph is more applicable for tracking problem. A very sparse measurement matrix has been used to efficiently extract compressive features for object representation. What is more, our method exploits temporal and spatial context information for tracking, which is very insensitive to appearance change. Experiments on some challenging video sequences have demonstrated the superiority of our proposed method to six state-of-the-art ones in accuracy and robustness.

7. REFERENCES

[1] Alper Yilmaz, Omar Javed, and Mubarak Shah, "Object tracking: A survey," *Acm Computing Surveys (CSUR)*, vol. 38, no. 4, pp. 13, 2006.

[2] Amit Adam, Ehud Rivlin, and Ilan Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. IEEE, 2006, vol. 1, pp. 798–805.

[3] 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, 2008.

[4] Xue Mei and Haibin Ling, "Robust visual tracking using 1 minimization," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1436–1443.

[5] Tianzhu Zhang, Bernard Ghanem, Si Liu, and Narendra Ahuja, "Robust visual tracking via multi-task sparse learning," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2042–2049.

[6] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang, "Real-time compressive tracking," in *Computer Vision–ECCV 2012*, pp. 864–877. Springer, 2012.

[7] Keren Fu, Chen Gong, Yu Qiao, Jie Yang, and Irene Yu-Hua Gu, "One-class support vector machine-assisted robust tracking," *Journal of Electronic Imaging*, vol. 22, no. 2, pp. 023002–023002, 2013.

[8] Boris Babenko, Ming-Hsuan Yang, and Serge Belongie, "Robust object tracking with online multiple instance learning," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 8, pp. 1619–1632, 2011.

[9] Kaihua Zhang and Huihui Song, "Real-time visual tracking via online weighted multiple instance learning," *Pattern Recognition*, 2012.

[10] Jingrui He, Mingjing Li, Hong-Jiang Zhang, Hanghang Tong, and Changshui Zhang, "Manifold-ranking based image retrieval," in *Proceedings of the 12th annual ACM international conference on Multimedia*. ACM, 2004, pp. 9–16.

[11] Dengyong Zhou, Jason Weston, Arthur Gretton, Olivier Bousquet, and Bernhard Schölkopf, "Ranking on data manifolds," *Advances in neural information processing systems*, vol. 16, pp. 169–176, 2003.

[12] Bin Xu, Jiajun Bu, Chun Chen, Deng Cai, Xiaofei He, Wei Liu, and Jiebo Luo, "Efficient manifold ranking for image retrieval," in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, 2011, pp. 525–534.

[13] Chen Gong, Keren Fu, Artur Loza, Qiang Wu, Jia Liu, and Jie Yang, "Pagerank tracker: From ranking to tracking," 2013.

[14] Paul Viola and Michael Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. IEEE, 2001, vol. 1, pp. I–511.