Sparse vs. Non-sparse: Which One Is Better for Practical Visual Tracking?

Yashar Deldjoo[1]
yashar.deldjoo@polimi.it
Shengping Zhang[2]
shengping.zhang@gmail.com
Bahman Zanj[3]
zanj@guilan.ac.ir
Paolo Cremonesi[1]
paolo.cremonesi@polimi.it
Matteo Matteucci[1]
matteo.matteucci@polimi.it

[1] Via Ponzio 34/5
Politecnico di Milano
Milano, Italy

[2] 224 Waterloo Road,
Harbin Institute of Technology,
Hong Kong, China

[3] Faculty of Engineering
University of Guilan,
Rasht, Iran

Abstract

Recently, sparse representation based visual tracking methods have attracted increasing attention in the computer vision community. Although achieve superior performance to traditional tracking methods, however, a basic problem has not been answered yet — that whether the sparsity constrain is really needed for visual tracking? To answer this question, in this paper, we first propose a robust non-sparse representation based tracker and then conduct extensive experiments to compare it against several state-of-the-art sparse representation based trackers. Our experiment results and analysis indicate that the proposed non-sparse tracker achieved competitive tracking accuracy with sparse trackers while having faster running speed, which support our non-sparse tracker to be used in practical applications.

1 Introduction

Visual tracking, *i.e.*, tracking a specific target object in consecutive video frames to get its moving trajectory, is one of the most important tasks in computer vision. A wide range of applications rely on robust visual tracking including, security and surveillance [10, 13], vehicle transportation and traffic monitoring [2, 5, 12], video compression [8, 18, 22], head-tracking, gesture recognition and eye-gaze tracking [1, 6, 7, 20]. Visual tracking has been extensively studied in the past decades in the computer vision community; however, it is still very challenging to handle irregular appearance changes of the tracked object during tracking, which are mainly due to abrupt geometric transformation, photometric variations like sudden change in illumination, and partial or full occlusions.

In the literature, a large number of tracking approaches have been proposed which can be roughly grouped in two main classes: *discriminative* methods and *generative* methods. The former formulates the tracking problem as the binary classification of distinguishing

the object from its background while the latter builds an appearance model of the target and formulates the tracking problem as a matching problem. Recently, inspired by the success of sparse representation in face recognition [24], sparse coding [19] has been successfully used in visual tracking [11, 16, 28, 29, 30]. Among them, ℓ_1 minimization based tracking method [16] formulates visual tracking as a reconstruction problem in a linear space where it is reasonable to impose a *sparse* constrain on the representation coefficients that the tracked target should be linearly represented by a small set of target templates with small reconstruction error. To make the tracker robust to occlusions, a set of occlusion templates are used in the linear representation to handle occlusions. Since this pioneer work, several researchers have tried to improve it by constraining the activation of these extra templates to improve the tracking accuracy [4] or by reducing the dimension of space to reduce tracking computational complexity [14, 26]. In the following part of the paper, we name all ℓ_1 minimization based trackers as ℓ_1 trackers.

Although promising results were reported at the time [16] was written and even though a number of other works have applied ℓ_1 trackers in their specific contexts [11, 28, 29, 30], the real role of the sparse constrain in the sparse representation was not well investigated in videos containing a variety of tracking circumstances. In particular, several studies in object recognition [21, 27] have experimentally indicated non-sparse representation with ℓ_2 norm minization has gotten superior performance than sparse representation with ℓ_1 norm mimization. Therefore, it is also necessary to investigate the roles of sparsity in ℓ_1 trackers. In addition, ℓ_1 trackers are inevitably computationally expensive due to their iterative optimization procedure. Most ℓ_1 trackers neglect the real-time requirement, which is very important for practical applications. In this paper, we aim at answering a basic question in ℓ_1 trackers that whether sparsity is really needed for visual tracking. To this aim, we first propose a non-sparse tracker and then conduct extensive experiments to compare it against several sparse trackers. Our experiment results and analysis indicate that the proposed non-sparse tracker has achieved competitive tracking accuracy while having faster running speed, which is better than sparse trackers for practical applications.

The rest of the paper is organized as it follows. Section 2 first review the existing sparse trackers. In Section 3, the proposed non-sparse tracker is introduced in detail. Experiments are reported and analyzed in Section 4. Section 5 concludes the paper.

2 Sparse tracker

Inspired by the success of sparse representation in face recognition [24], Mei *et al.* first proposed to model visual tracking as a sparse reconstruction problem under particle filter framework [16] . In particular, let $\mathbf{y} \in \mathbb{R}^d$ be a feature vector obtained by stacking the pixel intensities of a target candidate into a column vector and $\mathbf{T} = [\mathbf{t}_1\ \mathbf{t}_2\ \ldots \mathbf{t}_n] \in \mathbb{R}^{d\times n}$ be the set of feature vectors of previous target templates, which is manually collected at the first frame and then updated in an online fashion over time. It is natural to assume that the target templates $\mathbf{T}$ should span a linear space where the candidate is in. Formally, the target candidate $\mathbf{y}$ is represented in the following linear combination

$$\mathbf{y} = \alpha_1\mathbf{t}_1 + \alpha_2\mathbf{t}_2 + \ldots + \alpha_n\mathbf{t}_n + \eta = \mathbf{T}\alpha + \eta \tag{1}$$

where the templates $\mathbf{T}$ constructs the sparse representation dictionary, $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_n]^\top \in \mathbb{R}^n$ is the coefficient vector and $\eta \in \mathbb{R}^d$ is the noise term. To handle occlusion, a set of

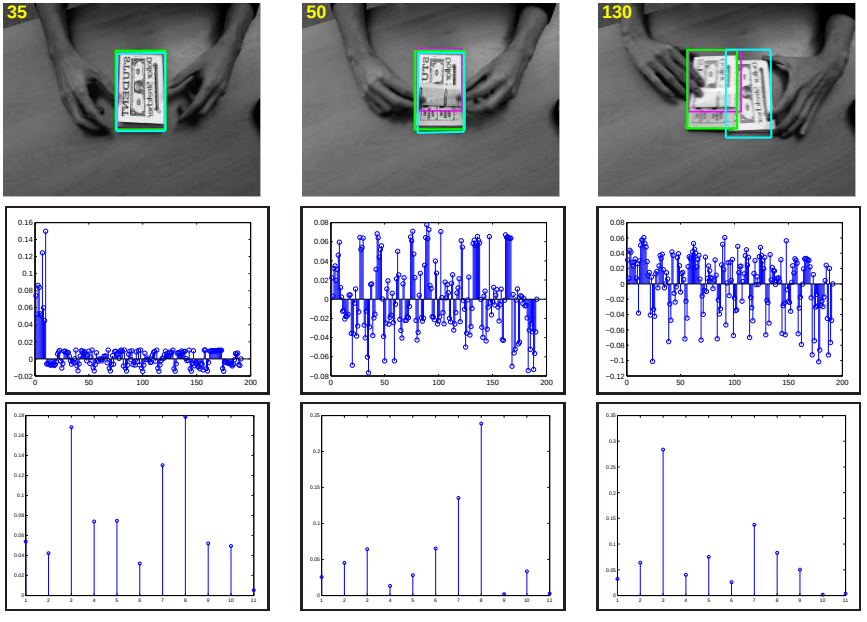


Figure 1: Comparing visual tracking algorithms using different dictionary types, row 1: sample tracking frames - (green) baseline, (cyan) tracker using augmented template dictionary, (magenta) tracker using basis template dictionary, row 2: The solution coefficients for tracker cyan, row 3: The solution coefficients for tracker magenta.

occlusion templates $\mathbf{I} = [\mathbf{i}_1, \mathbf{i}_2, \ldots, \mathbf{i}_d] \in \mathbb{R}^{d \times d}$ is further introduced into the dictionary and the final linear combination is defined as

$$\mathbf{y} = \begin{bmatrix} \mathbf{T} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \alpha \\ \mathbf{e} \end{bmatrix} = \mathbf{D}\mathbf{c} \tag{2}$$

where a occlusion template $\mathbf{i}_i \in \mathbb{R}^d$ is a vector with only one nonzero entry (*i.e.* $\mathbf{I}$ is an identity matrix), $\mathbf{D} = \begin{bmatrix} \mathbf{T} & \mathbf{I} \end{bmatrix} \in \mathbb{R}^{d \times (n+d)}$ is the augmented overcomplete dictionary, $\mathbf{c} = \begin{bmatrix} \alpha \\ \mathbf{e} \end{bmatrix} \in \mathbb{R}^{n+d}$ is the augmented coefficient vector and $\mathbf{e} = [e_1, e_2, \ldots, e_d]^\top \in \mathbb{R}^d$ is the occlusion coefficient vector.

When assuming it to be sparse, the coefficient vector $\mathbf{c}$ can be obtained by solving the following ℓ_1 minimization problem

$$\hat{\mathbf{c}} = \arg\min_{\mathbf{c}} \|\mathbf{y} - \mathbf{D}\mathbf{c}\|_2^2 + \lambda \|\mathbf{c}\|_1 \tag{3}$$

where the first and second terms measure the reconstruction error and the sparsity of the coefficient vector, respectively, and λ is a constant that controls the importance of the reconstruction error to the sparsity. Once the coefficient vector is obtained, the tracking result is found as the target candidate with the smallest reconstruction error after projecting on the target template subspace, *i.e.*, $\|\mathbf{y} - \mathbf{T}\alpha\|_2^2$.

Although the desired performance was reported, especially the robustness to occlusion, there are several major drawbacks. Firstly, the sparse assumption on the coefficients

may not hold in practice. In image classification field, several research studies [21, 27] have indicated that non-sparse representation such as collaborative representation achieves the competitive classification performance with sparse representation. It is also necessary to investigate whether sparsity is really needed for visual tracking. Second, solving the ℓ_1 minimization problem (Eq. 3) is very time-consuming, which restricts the tracker being used in real-time. Thirdly, the choice of occlusion templates is built upon a holistic idea to handle occlusion. In Figure 1, we show an example how the use of an augmented dictionary containing occlusion templates can lead to target loss. The dollar notes have a similar appearance to the target note on top and as the person starts folding the note (frame 50) and moving it to the left (frame 130), in the model using augmented dictionary $\mathbf{D} = [\mathbf{T}, \mathbf{I}]$ we can see a large occlusion templates activated (*i.e.* the coefficients become non-sparse) leading to the target loss whereas in simpler model with only basis target templates, the tracker learns the variation of the appearance in the target without trying to represent the difference via the help of occlusion templates as in the first case. We can conclude that the notion of occlusion templates to represent occlusion is built upon a holistic idea and in some cases it can lead to mis-classification of the target with its surrounding objects or background. To overcome the above drawbacks, several works have improved the work of [16]. For example, Mei *et al.* [17] proposed to reduce the number of ℓ_1-minimization by first sorting out the candidates based on their least-square residual error and accepting only candidates above a minimal threshold error to building a linear appearance model. Li *et al.* [14] and Zhang et al. [26] both made use of compressive sensing to build tracking models with real-time performance. An interesting work was proposed by Bao *et al.* [4] in which the authors proposed a real-time ℓ_1-tracker with improved tracking accuracy. The algorithm, which we shall revisit and refer to it as L1-APG, gains accuracy improvement via building a new minimization model for finding sparse representation of the target and real-time performance by a new APG (Accelerated Proximal Gradient) based numerical solver for resulting ℓ_1 problem.

3 Non-Sparse ridge regression based tracker

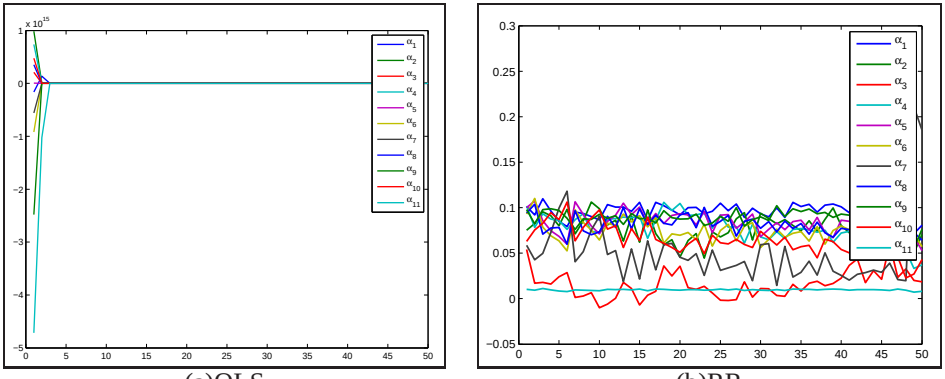
In this paper, we propose a robust non-sparse tracker based on ridge regression (RR). Instead of augumenting occlusion templates in the dictionary, here we only use the target templates $\mathbf{T} \in \mathbb{R}^{d \times n}$ as the dictionary. The basic ordinary least square (OLS) for computing the coefficients is given by

$$\hat{\alpha}_{OLS} = \arg\min_{\alpha} \|\mathbf{y} - \mathbf{T}\alpha\|_2^2 \tag{4}$$

which has a least square approximation solution

$$\hat{\alpha}_{OLS} = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{y} \tag{5}$$

Often, as in visual tracking, there is a linear dependency between two or more columns of $\mathbf{T}$ which causes to the precision of OLS become very poor. The columns in this case are called *multi-colinear* and may occur in two forms: (1) Exact multi-colinearity: the matrix $\mathbf{T}$ is singular. (2) Near multi-colinearity: at least one of the eigenvalues of the grammian matrices $\mathbf{T}^T\mathbf{T}$ or $\mathbf{T}\mathbf{T}^T$ is very small. In this condition, the linear system obtained becomes ill-conditioned and prohibits us from deriving a reliable linear representation. In such



(a)OLS (b)RR

Figure 2: Comparing solution coefficients between OLS and RR models. Heed to the difference of scales for y-axes in two cases which is extremely larger for OLS. After some frames the tracker using OLS drifts and the coefficients estimated become invalid.

condition, a reasonable remedy can be obtained through ℓ_2-regularization

$$\hat{\alpha}_{ridge} = \underset{\alpha}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{T}\alpha\|_2^2 + \lambda_{ridge} \|\alpha\|_2^2 \tag{6}$$

where λ_{ridge} is a constant regulatory parameter that makes a trade-off between the reconstruction error and the energy of coefficients. RR admits a direct analysis solution given by Eq. (7)

$$\hat{\alpha}_{ridge} = (\mathbf{T}^T\mathbf{T} + \lambda_{ridge}\mathbf{I})^{-1}\mathbf{T}^T\mathbf{y} \tag{7}$$

where $I \in \mathbb{R}^{n\times n}$ is the identity matrix. In statistics, Eq. (6) is known as *ridge regression* (RR) and was first introduced by Hoerl and Kennard [9]; in vision community it is also known as *collaborative representation* (CR). To demonstrate the effects of this condition on the estimation of coefficient, we consider sum of coefficients variances (total variance) for $\hat{\alpha}_{OLS}$ and $\hat{\alpha}_{ridge}$ which is given by

$$TV(\hat{\alpha}_{OLS}) = \sigma^2 . \sum_{j=1}^{s} \frac{1}{\lambda_j} \tag{8}$$

in which λ_j is the j-th eigenvalue of $\mathbf{T}$. It can be seen that total variance of OLS would be severely inflated if one or more columns are co-linear. For RR, Eq. (8) becomes

$$TV(\hat{\alpha}_{ridge}) = \sigma^2 . \sum_{j=1}^{s} \frac{\lambda_j}{(\lambda_j + \lambda_{ridge})^2} \tag{9}$$

By comparing Eq. (8) and Eq. (9), it can be noted that for any $\lambda_{ridge} > 0$, RR has a smaller total variance compared to OLS. In Figure 2, we compare the estimated solution coefficients for a randomly selected video under the OLS and RR which could be seen the coefficients are extremely unstable for OLS (in the range of 10^{15}) which is by far larger than RR with stabilized coefficients. As for related works, Zhang *et al.* [27] showed that great face recognition results reported by [24] were not achieved necessarily on the sparsity constraint and reported competitive results with collaborative representation which replaced ℓ_1-norm regularization with ℓ_2-norm regularization in sparse representation model.

The advantage of this model was suggested as a simple yet efficient solution compared to sparse representation as the optimization model admits a direct and efficient analytic solution. Li *et al.* [15] proposed a non-sparse based tracker that used a Mahalanobis distance metric (instead of Euclidean distance) for classification. The drawback of their approaches approach was the estimation of the weight matrix accurately which can be slow for visual tracking for which the authors proposed learning the weight matrix in an online fashion.

4 Experiments

4.1 Experimental setup

The proposed RR-based tracker (with ℓ_2-norm penalization) is compared against three state-of-the-art sparse trackers based on ℓ_1-norm penalization including L1-APG [4] (Accelerated Proximal Gradient), L1-WMB [17] (With Minimum Bound) and L1-Original [16]. Their underlying working characteristics are compared in Table 1. The main differences between the RR-based tracker and the compared trackers are in the complexity of the dictionary they use (*i.e.*, $\mathbf{T}$: basis template dictionary versus $\mathbf{D} = [\mathbf{T}, \mathbf{I}]$: augmented dictionary) and in the optimization model they are built upon (*i.e.* ℓ_1 vs. ℓ_2). It is important to point out the following remarks: (1) The sparse ℓ_1 trackers all use an augmented dictionary; (2) Our proposed non-sparse RR-based tracker does not use the occlusion dictionary because only under the ℓ_1-norm context that promotes sparsity, the use of occlusion dictionary was suggested to be useful for handling occlusion and such a judgment cannot be made in the ℓ_2-norm context.

We conducted extensive quantitative experiments on a total of 33 video sequences which are diverse and contain variety of tracking challenges. These video sequences are collected from the large scale benchmark library presented in [25] as well as [3]. For all trackers, we provide quantitative evaluation criterion defined by the Center Location Error (CLE) and Tracking Success Rate (TSR) which are computed based on a given ground truth. Since the trackers can have a dependency on the random number generation (RNG), we set the seed for the RNG to a fixed non-negative value which would allow us to have a fair comparison between all trackers under similar conditions. Furthermore, in order not be biased to only one realization of random numbers, without reinitialization from the same seed we obtain a sequence of random numbers and run each tracking algorithm 10 times on each video sequence according to the same random number. Results are reported in terms of the average ($\overline{CLE} = \frac{1}{N} \cdot \sum_{i=1}^{N} CLE_i$, $\overline{TSR} = \frac{1}{N} \cdot \sum_{i=1}^{N} TSR_i$) and standard deviation of the results obtained where $N = 10$ is the number of evaluation and

Table 1: Characteristics of the compared trackers obtained from [23] and modified. PF: Particle Filter

Tracker	Dictionary	Appearance model	Motion model	Optimization method	Update mechanism
Proposed RR	$\mathbf{D} = \mathbf{T}$	linear represen-taion intensities	Gaussian, PF	ℓ_2- regularization	Update bounding boxes, cosine similarity
L1-APG [4]	$\mathbf{D} = [\mathbf{T}, \mathbf{I}]$	linear represen-taion, intensities	Gaussian, PF	ℓ_1-regularization, constrained particles	Update bounding boxes, cosine similarity
L1-WMB [17]	$\mathbf{D} = [\mathbf{T}, \mathbf{I}]$	linear represen-taion, intensities	Gaussian, PF	ℓ_1-regularization, constrained particles	Update bounding boxes, cosine similarity
L1-Original [16]	$\mathbf{D} = [\mathbf{T}, \mathbf{I}]$	linear represen-taion, intensities	Gaussian, PF	ℓ_1-regularization	Update bounding boxes, cosine similarity

Table 2: Comparison of ours vs. three state-of-the-art approaches based on average center location error (CLE). The results in bold are significantly different with an α-confidence level of 5%

No.	Seq	Proposed RR	rank	L1-APG	rank	L1-WMB	rank	L1-Original	rank
1	Car4	**6.78 ± 0.27**	1	**6.59 ± 0.40**	1	111.50 ± 19.98	2	115.90 ± 17.87	2
2	CarDark	**13.51 ± 4.85**	1	**14.77 ± 5.59**	1	36.65 ± 12.05	2	46.82 ± 21.27	2
3	CarScale	**12.62 ± 2.7**	1	**15.79 ± 1.98**	1	47.37 ± 24.42	2	62.37 ± 35.93	2
4	Cliffbar	**3.76 ± 2.44**	1	**5.83 ± 1.36**	1	10.74 ± 2.62	2	12.11 ± 2.94	2
5	Coke	**27 ± 25.19**	1	123.8 ± 31.75	2	118.9 ± 13.75	2	124.6 ± 11.46	2
6	Couple	**18.87 ± 2.55**	1	**32.67 ± 16.11**	1	60.69 ± 24.76	2	78.66 ± 24.06	2
7	Crossing	**1.81 ± 0.27**	1	**7.1 ± 16.17**	1	**2.1 ± 0.08**	1	**2.06 ± 0.14**	1
8	David2	**3.96 ± 6.14**	1	**3.71 ± 1.52**	1	64.14 ± 5.77	2	58.26 ± 16.39	2
9	David	**8.37 ± 2.89**	1	**11.29 ± 5.69**	1	**7.92 ± 3.73**	1	**7.35 ± 4.08**	1
10	Deer	**6.56 ± 0.75**	1	**35.99 ± 36.68**	1	100.37 ± 42.5	2	85.84 ± 29.04	2
11	Doll	**4.37 ± 0.24**	1	**3.73 ± 0.79**	1	48.26 ± 24.29	2	62.73 ± 30.19	2
12	Dollar	**2.29 ± 0.65**	1	13.42 ± 0.25	2	14.41 ± 3.4	2	13.56 ± 0.24	2
13	Dudek	**10.92 ± 1.54**	1	93.12 ± 62.75	2	136.1 ± 52.31	2	101.2 ± 43.77	2
14	FaceOcc1	**17.11 ± 2.29**	1	**15.04 ± 0.67**	1	75.62 ± 59.19	2	**54.79 ± 53.44**	1
15	FaceOcc2	**12.32 ± 3.66**	1	**13.75 ± 1.23**	1	**29.74 ± 36.29**	1	**29.9 ± 36.74**	1
16	Fish	**43.21 ± 16.87**	1	**41.44 ± 22.61**	1	71.1 ± 29.66	2	**58.77 ± 25.66**	1
17	FleetFace	**20.15 ± 2.97**	1	41.33 ± 21.68	2	114.88 ± 6.53	3	114.99 ± 11.4	3
18	Football1	**17.81 ± 9.52**	1	**28.01 ± 13.86**	1	61.35 ± 9.65	2	58.45 ± 9.91	2
19	Football	**13.81 ± 1.07**	1	60.86 ± 54.28	2	96.35 ± 20.72	3	112.5 ± 11.44	3
20	Freeman1	**61.03 ± 39.38**	1	**45.32 ± 33.13**	1	**32.21 ± 29.98**	1	**55.8 ± 23.9**	1
21	Freeman3	**7.28 ± 4.03**	1	26.69 ± 10.97	2	55.22 ± 2.73	2	38.53 ± 22.28	3
22	Freeman4	**22.66 ± 18.07**	1	**30.18 ± 18.7**	1	57.97 ± 14.67	2	60.13 ± 14.76	2
23	Girl	**5.29 ± 3.05**	1	**7.39 ± 3.04**	1	**16.58 ± 14.63**	1	**13.94 ± 13.97**	1
24	Jumping	**16.47 ± 22.75**	1	**4.61 ± 0.6**	1	**27.58 ± 41.49**	1	**34.74 ± 36.81**	1
25	Mhyang	**2.62 ± 0.73**	1	**3.01 ± 1.23**	1	36.27 ± 6.14	2	38.32 ± 4.51	2
26	Mountain Bike	**8.16 ± 1.98**	1	160.3 ± 78.48	2	201.5 ± 57.65	2	207.6 ± 59.65	2
27	Singer1	**4.34 ± 0.74**	1	**4.65 ± 0.75**	1	**54.39 ± 80.43**	1	**21.86 ± 53.92**	1
28	Soccer	**60.83 ± 14.44**	1	**88.56 ± 36.8**	1	152.6 ± 32.32	2	133.5 ± 27.75	2
29	Subway	36.32 ± 1.06	2	37.06 ± 1.49	2	**3.97 ± 0.17**	1	**4.01 ± 0.22**	1
30	Surfer	**1.9 ± 0.67**	1	13.38 ± 0.24	2	13.59 ± 0.42	2	14.6 ± 3.84	2
31	SUV	**26.38 ± 28.43**	1	51.15 ± 23.62	2	**26.17 ± 2.13**	1	**26.46 ± 2.53**	1
32	Sylvester	**18.51 ± 9.94**	1	**32.66 ± 11.49**	1	**44.85 ± 28.54**	1	51.12 ± 31.26	2
33	Trellis	**12.95 ± 8.15**	1	**31.66 ± 7.45**	1	65.71 ± 22.82	2	81.03 ± 34.29	2
Avg $\overline{CLE}$/rank		**16.06**	**1.03**	33.48	1.30	60.51	1.76	60.08	1.76

CLE_i and TSR_i are the average CLE and TSR over the entire frames in each run. The parameters related to the particle file variance parameters in our experiment were set to be like in the benchmark $[0.03, 0.0005, 0.0005, 0.03, 1, 1]$ where $\mathbf{t}_x$ = $\mathbf{t}_y$ = 1 are translations in **x** and **y** directions. In some videos containing fast motion or pose change (*e.g.* CarDark, Coke, Deer etc.) the variances were changed correspondingly to for example $[0.03, 0.0005, 0.0005, 0.03, 2, 2]$ to be able to capture the fast motions, the same for all tracker. The regulatory parameter for ℓ_1 trackers were used as they were used by the original codes. The regulatory parameter for our proposed algorithm was set to $\lambda_{ridge} = 1$ or $\lambda_{ridge} = 2$ in most videos which resulted in fairly similar performance. In rare cases containing severe occlusion (e.g. Coke) λ_{ridge} was increased to higher values which had a positive effect because it avoided frequent update of the dictionary and insertion of bad template in the dictionary.

4.2 Significance Testing for Quantitative Evaluation

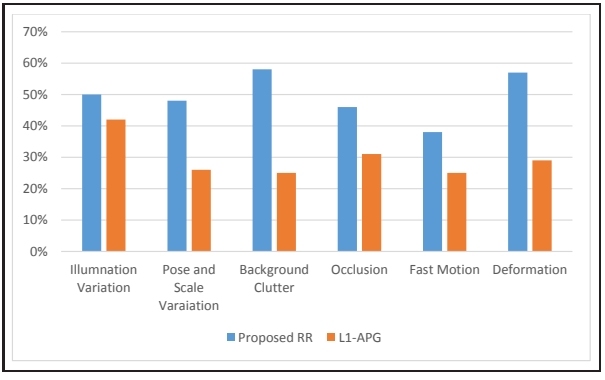
Table 2 and Table 3 present the computed $\overline{CLE}$ and $\overline{TSR}$ for the all compared trackers on each test sequence. A multiple pairwise comparison testing based on one-way analysis of variance (ANOVA) is applied on each video sequence to evaluate whether or not the

Table 3: Comparison of ours vs. three state-of-the-art approaches based on tracking success rate (TSR). The results in bold are significantly different with an α-confidence level of 5%.

No.	Seq	Proposed RR	rank	L1-APG	rank	L1-WMB	rank	L1-Original	rank
1	Car4	**1 ± 0**	1	**1 ± 0**	1	0.14 ± 0.14	2	0.09 ± 0.13	2
2	CarDark	**0.74 ± 0.09**	1	**0.72 ± 0.1**	1	**0.52 ± 0.18**	1	0.34 ± 0.29	2
3	CarScale	**0.83 ± 0.07**	1	**0.79 ± 0.01**	1	0.6 ± 0.2	2	0.47 ± 0.3	2
4	Cliffbar	**0.43 ± 0.1**	1	**0.4 ± 0.05**	1	0.3 ± 0.13	2	0.24 ± 0.13	2
5	Coke	**0.67 ± 0.25**	1	0.06 ± 0.01	2	0.03 ± 0.01	2	0.02 ± 0.01	2
6	Couple	**0.52 ± 0.07**	1	**0.44 ± 0.09**	1	0.21 ± 0.14	2	0.11 ± 0.15	2
7	Crossing	**0.95 ± 0.04**	1	**0.88 ± 0.23**	1	**0.98 ± 0.04**	1	**0.97 ± 0.02**	1
8	David2	**0.93 ± 0.14**	1	**0.83 ± 0.11**	1	0.15 ± 0.08	2	0.23 ± 0.14	2
9	David	**0.22 ± 0.01**	1	**0.22 ± 0.01**	1	**0.21 ± 0.01**	1	**0.22 ± 0.02**	1
10	Deer	**0.99 ± 0.02**	1	**0.78 ± 0.24**	1	0.39 ± 0.28	2	0.44 ± 0.28	2
11	Doll	0.51 ± 0.17	2	**0.81 ± 0.16**	1	0.54 ± 0.14	2	0.42 ± 0.19	2
12	Dollar	**0.92 ± 0.17**	1	0.39 ± 0	2	0.35 ± 0.12	2	0.37 ± 0.02	2
13	Dudek	**0.78 ± 0.03**	1	**0.68 ± 0.09**	1	0.31 ± 0.31	2	0.51 ± 0.26	2
14	FaceOcc1	**0.91 ± 0.1**	1	**0.98 ± 0.02**	1	0.45 ± 0.46	2	**0.6 ± 0.41**	1
15	FaceOcc2	**0.42 ± 0.08**	1	**0.38 ± 0.03**	1	0.27 ± 0.14	2	**0.3 ± 0.15**	1
16	Fish	**0.05 ± 0.01**	1	**0.14 ± 0.14**	1	**0.07 ± 0.06**	1	**0.09 ± 0.06**	1
17	FleetFace	**0.64 ± 0.01**	1	**0.64 ± 0.02**	1	0.52 ± 0.02	2	0.52 ± 0.01	2
18	Football1	**0.57 ± 0.16**	1	0.32 ± 0.1	2	0.12 ± 0.05	3	0.15 ± 0.08	3
19	Football	**0.7 ± 0.06**	1	0.45 ± 0.12	2	0.14 ± 0.07	3	0.12 ± 0.08	3
20	Freeman1	**0.19 ± 0.04**	1	**0.17 ± 0.05**	1	**0.23 ± 0.11**	1	**0.22 ± 0.04**	1
21	Freeman3	**0.71 ± 0.15**	1	0.59 ± 0.09	2	0.59 ± 0.01	2	**0.6 ± 0.06**	1
22	Freeman4	**0.3 ± 0.1**	1	**0.35 ± 0.11**	1	0.24 ± 0.02	2	0.24 ± 0.04	2
23	Girl	**0.67 ± 0.15**	1	**0.46 ± 0.16**	1	**0.48 ± 0.37**	1	**0.54 ± 0.35**	1
24	Jumping	**0.71 ± 0.25**	1	**0.94 ± 0.06**	1	**0.7 ± 0.37**	1	0.55 ± 0.38	2
25	Mhyang	**0.99 ± 0.03**	1	**0.98 ± 0.03**	1	0.6 ± 0.09	2	0.54 ± 0.12	2
26	Mountain Bike	**0.74 ± 0.16**	1	0.37 ± 0.12	2	0.06 ± 0.05	3	0.06 ± 0.05	3
27	Singer1	**0.97 ± 0.11**	1	0.96 ± 0.07	2	0.69 ± 0.47	2	0.85 ± 0.3	2
28	Soccer	**0.18 ± 0.02**	1	**0.15 ± 0.02**	1	0.05 ± 0.04	2	0.08 ± 0.03	2
29	Subway	**0.5 ± 0.02**	1	**0.5 ± 0.01**	1	0.89 ± 0.03	2	0.88 ± 0.03	2
30	Surfer	**1 ± 0**	1	0.39 ± 0	2	0.38 ± 0.02	2	0.35 ± 0.12	2
31	SUV	**0.85 ± 0.16**	1	0.65 ± 0.12	2	0.69 ± 0	2	0.69 ± 0	2
32	Sylvester	**0.65 ± 0.12**	1	0.43 ± 0.06	2	0.24 ± 0.12	3	0.21 ± 0.14	3
33	Trellis	**0.51 ± 0.13**	1	0.32 ± 0.09	2	0.19 ± 0.12	2	0.18 ± 0.13	2
Avg $\overline{TSR}$/rank		**0.66**	**1.03**	0.55	1.33	0.37	1.82	0.37	1.88

difference between the groups' averages for each tracker most likely reflects a significant difference or not. ANOVA is a generalized significance t-test which is applicable when the test statistic would follow a normal distribution. We argue that the normality assumption can be made on the groups' distributions based on *the central limit theorem*. The reason is e.g. for CLE_i (the i-th evaluation of CLE), it is the average of many random variables (errors at each pixel) which can be assumed to be independent and identically distributed (i.i.d). Central limit theorem therefore states that the mean of these i.i.d random variables (*i.e.* CLE_i) follows a normal distribution and therefore a significance t-test is applicable on the group of CLE_i's obtained from different evaluations. The result of such ANOVA-based significance test is provided in Tables 2 and Table 3 as a ranking value on each video which implies based on α-level significance test ($\alpha = 5\%$), if two algorithms have significantly different performance or not. For example in the video Deer, the computed $\overline{CLE}$ for RR and L1-APG are 6.56 ± 0.75 and 35.99 ± 36.68 respectively. While the absolute value of the $\overline{CLE}$'s are greatly different, their α-level significance test show that they are not significantly different and thus they are both given the same ranking equal to 1. In another video Dollar for instance, the $\overline{CLE}$ of RR and L1-APG with 2.29 ± 0.65 and 13.42 ± 0.25

Figure 3: Comparing the performance of two competing trackers in handling different challenging tracking scenarios during the tracking process



based on the significance test are considered significantly different. In this manner, we are able to conduct a fair comparison between the performance of trackers based on the results obtained from different evaluations.

4.3 Comparison of competing trackers

The performance of the proposed RR-based tracker against the competing ℓ_1 trackers can be compared in Table 2 and Table 3 according to the average rankings computed by averaging out the rankings computed based on α-level significance test on each video. As could be seen our proposed RR-based tracker has the best ranking (*i.e.* 1.03 and 1.03) against the competing trackers which shows it is capable of effectively handling complicated appearance changes in the tracking process. In Figure 3, we also provide the performance of the two best competing trackers under different tracking challenges as a means to compare their performances under such circumstances. The vertical axis is the percentage of videos with a particular challenge for which RR and L1-APG trackers pass it successfully. The challenges for each video were obtained found [25].As could be seen, RR outperforms L1-APG almost in all challenges. It could be as noted that both trackers are weak in handling fast motions which is the drawback of these trackers. Finally, the efficiency of the proposed tracker against the competing trackers in terms of average speed is compared in Table 4 and the results are greatly in favor of the proposed RR-based tracker.

Table 4: Comparison of the proposed vs. three state-of-the-art approaches based on average running speed in terms of frames/sec. The first best result is labeled by bold.

Seq	Proposed RR	L1-APG	L1-WMB	L1-Original
Avg Speed	**10.34**	4.85	3.09	3.1

5 Conclusion

Before the stress on sparsity and using complex dictionaries for handling occlusions etc., we have shown in this paper that the main problem in visual tracking arises from colinearity of data which could be solved by classical ridge regression. Indeed, too much push on sparsity leads to penalization of results with respects to classical ridge regression. To this end, a robust visual tracker based on non-sparse linear representation was proposed that can effectively handle different tracking challenges in extended tracking sequences. The results indicate that our proposed tracker can archive competitively better results compared to ℓ_1 trackers while having faster running speed, which supports the effectiveness of our proposed non-sparse tracker for practical applications.

References

[1] Amer Al-rahayfeh and Miad Faezipour Member. Eye Tracking and Head Movement Detection : A State-of-Art Survey. *IEEE Journal of Translational Engineering in Health and Medicine*, (August), 2013.

[2] S. Atev, H. Arumugam, O. Masoud, R. Janardan, and N. P. Papanikolopoulos. A vision-based approach to collision prediction at traffic intersections. *Trans. Intell. Transport. Sys.*, 6(4):416–423, December 2005. ISSN 1524-9050. doi: 10.1109/TITS.2005.858786. URL http://dx.doi.org/10.1109/TITS.2005.858786.

[3] Boris Babenko and Ming-Hsuan Yang Serge Belongie. Robust object tracking with online multiple instance learning. 2011.

[4] Chenglong Bao, Yi Wu, Haibin Ling, and Hui Ji. Real time robust l1 tracker using accelerated proximal gradient approach. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1830–1837. IEEE, 2012.

[5] Benjamin Coifman, David Beymer, Philip Mclauchlan, and Jitendra Malik. A real-time computer vision system for vehicle tracking and surveillance. *Transportation Research Part C*, 6:271–288, 1998.

[6] Yashar Deldjoo. *Wii remote based head tracking in 3D audio rendering*.

[7] Yashar Deldjoo and Reza Ebrahimi Atani. A low-cost infrared-optical head tracking solution for virtual 3d audio environment using the nintendo wii-remote. *Entertainment Computing*, 12:9–27, 2016.

[8] K. Hariharakrishnan and D. Schonfeld. Fast object tracking using adaptive block matching. *Trans. Multi.*, 7(5):853–859, October 2005. ISSN 1520-9210. doi: 10.1109/TMM.2005.854437. URL http://dx.doi.org/10.1109/TMM.2005.854437.

[9] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[10] Weiming Hu, Tieniu Tan, Liang Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *Trans. Sys. Man Cyber Part C*, 34(3):334–352, August 2004. ISSN 1094-6977. doi: 10.1109/TSMCC.2004.829274. URL http://dx.doi.org/10.1109/TSMCC.2004.829274.

[11] Zhangjian Ji and Weiqiang Wang. Object tracking based on local dynamic sparse model. *Journal of Visual Communication and Image Representation*, 2015.

[12] V Kastrinaki, M Zervakis, and K Kalaitzakis. A survey of video processing techniques for traffic applications. *Image and Vision Computing*, 21(4):359–381, April 2003. ISSN 02628856. doi: 10.1016/S0262-8856(03)00004-0. URL http://linkinghub.elsevier.com/retrieve/pii/S026288560300

[13] In Su Kim, Hong Seok Choi, Kwang Moo Yi, Jin Young Choi, and Seong G. Kong. Intelligent visual surveillance: A survey. *International Journal of Control, Automation and Systems*, 8(5):926–939, October 2010. ISSN 1598-6446. doi: 10.1007/s12555-010-0501-4. URL http://link.springer.com/10.1007/s12555-010-0501-4.

[14] Hanxi Li, Chunhua Shen, and Qinfeng Shi. Real-time visual tracking using compressive sensing. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1305 –1312, 2011.

[15] Xi Li, Chunhua Shen, Qinfeng Shi, Anthony Dick, and Anton van den Hengel. Non-sparse linear representations for visual tracking with online reservoir metric learning. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1760–1767. IEEE, 2012.

[16] X. Mei and H. Ling. Robust visual tracking using L1 minimization. *Proceedings of the 12th International Conference on Computer Vision*, pages 1436–1443, 2009.

[17] Xue Mei, Haibin Ling, Yi Wu, Erik Blasch, and Li Bai. Minimum error bounded efficient L1 tracker with occlusion detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1257–1264, 2011.

[18] Joan L. Mitchell, William B. Pennebaker, Chad E. Fogg, and Didier J. Legall, editors. *MPEG Video Compression Standard*. Chapman & Hall, Ltd., London, UK, UK, 1996. ISBN 0412087715.

[19] Bruno A Olshausen et al. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996.

[20] Vladimir I. Pavlovic, Rajeev Sharma, and Thomas S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):677–695, July 1997. ISSN 0162-8828. doi: 10.1109/34.598226. URL http://dx.doi.org/10.1109/34.598226.

[21] R. Rigamonti, M.A. Brown, and V. Lepetit. Are sparse representations really relevant for image classification? *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1545–1552, 2011.

[22] T. Sikora. The mpeg-4 video standard verification model. *IEEE Trans. Cir. and Sys. for Video Technol.*, 7(1):19–31, February 1997. ISSN 1051-8215. doi: 10.1109/76.554415. URL http://dx.doi.org/10.1109/76.554415.

[23] Arnold W M Smeulders, Senior Member, Dung M Chu, Student Member, Rita Cucchiara, and Simone Calderara. Visual Tracking : An Experimental Survey. 36(7): 1442–1468, 2014.

[24] John Wright, Allen Y. Yang, Arvind Ganesh, S. Shankar Sastry, and Yi Ma. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(2):210–227, February 2009. ISSN 0162-8828. doi: 10.1109/TPAMI.2008.79. URL http://dx.doi.org/10.1109/TPAMI.2008.79.

[25] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online object tracking: A benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[26] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang. Real-time compressive tracking. pages 864–877, 2012.

[27] L. Zhang, M. Yang, and Feng X. Sparse representation or collaborative representation: Which helps face recognition? *International Conference on Computer Vision*, pages 471–478, 2011.

[28] Shengping Zhang, Hongxun Yao, Xin Sun, and Xiusheng Lu. Sparse coding based visual tracking: Review and experimental comparison. *Pattern Recogn.*, 46(7): 1772–1788, July 2013. ISSN 0031-3203. doi: 10.1016/j.patcog.2012.10.006. URL http://dx.doi.org/10.1016/j.patcog.2012.10.006.

[29] 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*, pages 2042–2049. IEEE, 2012.

[30] Wei Zhong, Huchuan Lu, and Ming-Hsuan Yang. Robust object tracking via sparsity-based collaborative model. In *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on*, pages 1838–1845. IEEE, 2012.