Robust and Fast Collaborative Tracking with Two Stage Sparse Optimization

Baiyang Liu[1,2,⋆], Lin Yang[2], Junzhou Huang[1], Peter Meer[3], Leiguang Gong[4], and Casimir Kulikowski[1]

[1] Department of Computer Science, Rutgers University, NJ, USA
[2] Deptartment of Radiology, UMDNJ-Robert Wood Johnson Medical School, NJ, USA
[3] Department of Electrical and Computer Engineering, Rutgers University, NJ, USA
[4] IBM T.J. Watson Research, NY, USA

Abstract. The sparse representation has been widely used in many areas and utilized for visual tracking. Tracking with sparse representation is formulated as searching for samples with minimal reconstruction errors from learned template subspace. However, the computational cost makes it unsuitable to utilize high dimensional advanced features which are often important for robust tracking under dynamic environment. Based on the observations that a target can be reconstructed from several templates, and only some of the features with discriminative power are significant to separate the target from the background, we propose a novel online tracking algorithm with two stage sparse optimization to jointly minimize the target reconstruction error and maximize the discriminative power. As the target template and discriminative features usually have temporal and spatial relationship, dynamic group sparsity (DGS) is utilized in our algorithm. The proposed method is compared with three state-of-art trackers using five public challenging sequences, which exhibit appearance changes, heavy occlusions, and pose variations. Our algorithm is shown to outperform these methods.

1 Introduction

Tracking is to estimate the state of the moving target in the coming observed sequences. This topic is interesting for many industrial applications, such as surveillance, traffic monitoring, vehicle navigation, video indexing, etc. Accurate tracking of a general object in a dynamic environment is difficult due to the following challenges [1,2]:

- Dynamic appearance changes due to illumination, rotation, and scaling;
- 3D pose variations and information loss due to the projection;
- Partial and full object occlusions;
- Complex background clutters;
- Similar objects from the same class which lead to landmark ambiguity.

⋆ This research is completed when the author is a research assistant in the Department of Radiology in the UMDNJ-Robert Wood Johnson Medicial School.

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part IV, LNCS 6314, pp. 624–637, 2010.

Current tracking techniques can be categorized as discriminative or generative methods. Discriminative methods formulate the tracking as a classification problem[3,4,5,6]. The trained classifier is used to discriminate the target from background and can be online updated during the tracking procedure [7,8]. The generative methods represent the target observations as an appearance model [9]. The tracking problem is formulated as searching for the region with the highest probability generated from the appearance model [10,11,12,13,14,15,16]. It was proposed to update the target appearance model incrementally for adapting to dynamic environmental changes and target appearance variations. Generative models and discriminative models are combined and a one step forward prediction based collaborative tracking are proposed in [17].

Recently, sparse representations have been utilized in many areas [18,19,20,21] and successfully applied for tracking [22]. The tracking problem is formulated as finding a sparse approximation in the template subspace Φ. For candidate sample y, the general sparse problem can be formulated as

$$x_0 = argmin_x ||x||_0 \text{ subject to } ||y - \Phi x|| < \epsilon \tag{1}$$

where $||.||_0$ denotes the zero norm which represents the number of nonzero components and ϵ is the level of reconstruction error. However, it is well known that the l_0 optimization problem is NP-hard and there is no efficient algorithm to find the global optimum solution other than exhaustive search.

One class of algorithms tries to seek the sparsest solution by performing basis pursuit (BP) based l_1 minimization as

$$x_1 = argmin_x \ ||y - \Phi x|| + \tau |x|_1 \tag{2}$$

using linear programming instead of l_0 minimization in (1) [23]. This method is applied to solve l_1 minimization with none-negative constraints in [22]. The results are found to be efficient and adaptive to appearance changes, especially occlusion. However, there are still several problems exist:

- It is computationally expensive for very high dimensional data, which makes it unsuitable to use advanced image features for fast tracking applications.
- The background pixels in the target templates do not lie on the linear template subspace. The scale of the reconstruction error from background pixels is often larger than that from the target pixels, which might affect the accuracy of the sparse representation. It is therefore more reasonable to build the target template subspace *from the pixels belonging to the object.*
- The non-negative constraints, although can provide very good results when there are outliers, are vulnerable to complete tracking failures if wrong templates are selected.
- Temporal correlation between target templates and spatial relations among adjacent image features are not considered.
- Since the sparse parameter τ in (2) has no physical meaning, it is therefore difficult to tune up the parameter.

We observed that the target can usually be represented by templates sparsely and only part of the features, which can discriminate the target and background, are necessary to identify the target. Motivated by [22], considering existing problems and our observations, we proposed a robust and fast tracking algorithm with two stage sparse optimization. The algorithm starts from feature selection by solving a dynamic group sparsity (DGS) [24] optimization problem. The DGS is then performed on the selected feature space for sparse reconstruction of the target. These two sparsity problems are optimized jointly and the final results are obtained by Bayesian inference. According to our knowledge, this is the first study reporting fast and robust tracking algorithm using *two stage sparsity optimization*. The contributions of this paper are:

- A unified online updated sparse tracking framework which is targeted to use very high dimensional image features.
- The location adjacent features and time adjacent target templates tend to be selected as a group in our sparse representation, which provides more robust tracking results.
- The sparse parameters do have physical meaning and therefore are easy to be tuned.
- The algorithm is efficient. It is at least three times faster than the most current literature on sparse representation based tracking.
- Pose variation, appearance changes, and heavy occlusions are handled in our algorithm.

The paper is organized as follows: The related work is explained in Section 2. The tracking algorithm using two stage sparsity is presented in Section 3. Section 4 presents the experimental results. Finally, Section 5 concludes the paper.

2 Related Work

As online learning, sparse representation and dynamic group sparsity are intensively used in our algorithm, in this section we will give a brief review. Online adaptive tracking method has been intensively investigated in the recent literature. Grabner et al [7] propose to update the feature selection incrementally using the training samples gathered from current tracking result, which may lead to potential target drifting because of accumulated errors. Semi-online boosting [25] was proposed to incrementally update the classifier using unlabeled and labeled data together to avoid the target drifting. Multiple Instance Learning boosting method (MILBoosting) [4] put all samples into bags and labels them with bag labels. The positive bag is required to contain at least one real positive, while the negative bags have only negative samples. The drifting problem is handled in their method since the true target included in positive bag is learned implicitly. The target is represented as a single online learned appearance model in incremental visual tracking (IVT) [14]. As single appearance model is argued to be not sufficient to present the target in a dynamic environment, multiple appearance models are also proposed to be incrementally learned during the

tracking in [26]. Online updating is proven to be an important step in adaptive tracking and is also used in our algorithm.

Sparse representation was introduced for tracking in [22]. The target candidate is represented as a linear combination of the learned template set composed of both target templates and the trivial template which has only one nonzero element. The assumption is that good target candidate can be sparsely represented by both the target templates and the trivial templates. This sparse optimization problem is solved as a l_1 minimization problem with nonenegative constraints.

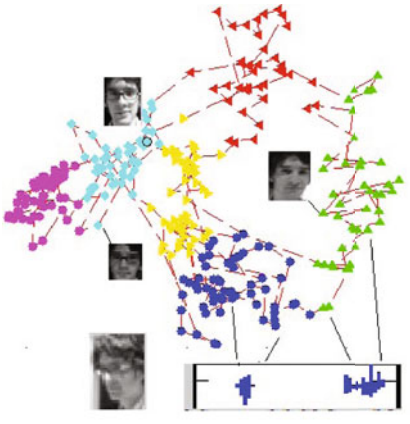

Fig. 1. The group structure of template feature vectors which can be clustered into six groups. The consecutive templates are connected with edges.

Another well known class of sparse optimization algorithms is the iterative greedy pursuit. The earliest algorithms including the matching pursuit [27] and orthogonal matching pursuit [28]. The subspace pursuit [29] and the compressive sampling matching pursuit [30] were proposed to reach similar theoretical recovery guarantees as the BP while reduce computational complexity. However, the nonzero components of the solution are not randomly distributed and tend to be clustered. Motivated by this prior, dynamic group sparsity (DGS) recovery algorithm is proposed in [24]. The algorithm includes five main steps in each iteration: 1) pruning the residue estimation; 2) merging the support sets; 3) estimating the signal by least square; 4) pruning the signal estimation and 5)updating the signal/ residue estimation and support set. The algorithm is similar to that of SP/CoSaMP [29,30] except considering the effect of neighbors also in the pruning process. DGS optimization also provides more robust result by forcing group representation which can eliminate wrong templates that do not fall in the same linear space as its neighbors. In Figure 1 we show the group structure of the consecutive learned templates in one of our testing tracking sequences. The image features are projected to two dimensional vector and clustered into six groups. In the bottom of Figure 1, we can tell that the target is sparsely represented by two groups. In other words,if one of the templates in the group is selected, its temporal adjacent templates tend to be selected too in our sparse representation using DGS.

3 Tracking with Two Stage Sparsity

We start this section from Bayesian tracking framework. The tracking algorithm is formulated as a two stage sparse optimization that is optimized jointly. The final results are obtained by Bayesian inference.

3.1 Bayesian Tracking Framework

Let affine parameters $\chi_t = (x, y, s, r, \theta, \lambda)$ represent the target state in the t-th frame, where x and y are the coordinates, s and r are the scale and the aspect, θ is the rotation angle, λ is the skew. The tracking problem can be formulated as an estimation of the state probability $p(\chi_t|z_{1:t})$, where z represents the observation in the previous t frames. Sequential Bayesian tracking based on Markovian assumption estimates and propagates the probability by recursively performing prediction

$$p(\chi_t|z_{1:t-1}) = \int p(\chi_t|\chi_{t-1})p(\chi_{t-1}|z_{1:t-1})d\chi_{t-1} \tag{3}$$

and updating

$$p(\chi_t|z_{1:t}) \propto p(z_t|\chi_t)p(\chi_t|z_{1:t-1}). \tag{4}$$

The transition model $p(\chi_t|\chi_{t-1})$ is constrained by assuming a Gaussian distribution $\mathcal{N}(\chi_t|\chi_{t-1}, \sigma)$. The observation model $p(z_t|\chi_t)$ represents the likelihood of z_t being generated from state χ_t.

In our algorithm, N candidate samples are generated based on the state transition model $p(\chi_t|\chi_{t-1})$. The state variables are considered as independent of each other. Each candidate sample I_i with state χ_t^i is reconstructed from the template library Φ using dynamic group sparsity (DGS). The likelihood $p(z_t|\chi_t^i) = exp(-\epsilon_i)$ where $\epsilon_i = min_\alpha||\Phi\alpha - I_i||$ is the optimized reconstruction error of I_i and α represents the sparse coefficients. Instead of solving the optimization problem in the full feature space, we propose to perform the sparse optimization in selected feature space with discriminative power. This enables us to use advanced high dimensional features without sacrificing the efficiency of the algorithm. Once the tracking state is confirmed, new samples are extracted and used to online update the training set and template library. The final result is obtained by maximizing $p(\chi_t|z_{1:t})$.

3.2 Two Stage Sparse Representation

Given the learned target template library $\Phi \in \mathbb{R}^{p\times m}$, where m is the number of templates and p is the dimension of the features. Let $\Phi_1 = [\Phi, I]$ and $\alpha_1 = \begin{bmatrix} \alpha \\ f \end{bmatrix}$ where α represents the sparse coefficient vector and f denotes the occlusion, the candidate sample y is sparsely reconstructed from Φ by minimizing the l_2 errors and finding α with K_1 nonzero components and f with K_2 nonzero components using greedy method:

$$\alpha_1 = argmin_{\alpha,f}\ ||\Phi_1\alpha_1 - y||_2, \text{while } ||\alpha||_0 \leq K_1 \text{ and } ||f||_0 \leq K_2. \tag{5}$$

Equation (5) can be solved efficiently when the dimension of the feature space and candidate searching space are small. However, it is computationally expensive for very high dimensional data, which make it unsuitable if advanced image

Algorithm 1. Tracking with two stage sparsity optimization

Input: Target's initial state χ_0, sparsity parameter K_0 for feature selection, K_1 and K_2 for target template and trivial template.

Initialize: Construct n training samples $\{X \in \mathbb{R}^{n\times p}, L \in \mathbb{R}^{n\times 1}\}$, where X is the sample matrix, L is the label and p is the dimension of the feature vector.

1. For each frame $t = 1 : T$ in the video where T is the total number of frames:
2. Perform DGS to solve $w^* = argmin_w ||Xw - L||_2$,
   subject to: $|w|_0 \leq K_0$ (when $t = 1$ we will use the initializations).
3. Construct diagonal matrix W, $W_{i,i} = \begin{cases} 1, w_i^* \neq 0 \\ 0, otherwise; \end{cases}$
4. Generate N candidate samples y_i in state χ_t^i.
5. For each $y_i, i = 1 : N$
6. Let $W' \in \mathbb{R}^{K_0\times p}$ as the matrix contains all non-zero rows of W,
7. $\Phi' = W'\Phi$, $y_i' = W'y_i$, and $f' = W'f$,
8. perform DGS to solve
9. $(\alpha^*, f^*) = argmin_{\alpha,f} \left|\left|[\Phi' \ W'] \begin{bmatrix} \alpha \\ f \end{bmatrix} - y_i'\right|\right|_2$, subject to: $\begin{matrix} ||\alpha||_0 \leq K_1 \\ ||f||_0 \leq K_2. \end{matrix}$
10. $\epsilon_i = ||\Phi'\alpha^* - y_i'||_2$.
11. $p(z_t|\chi_t^i) = exp(-\epsilon_i)$.
12. end for
13. $\chi_t^* = argmax_{\chi_t} p(\chi_t|z_{1:t})$.
14. Update the training set and template library with tracking results.
15. end for

features are used. Because only some of the features, which can discriminate the target and background, are necessary to identify the target, we argued that the effective dimension of the feature space can be decreased to K_0 dimension with diagonal matrix W. The number of nonzero components in W is not larger than K_0. The i-th feature is activated if W_{ii} is nonzero. Given n available samples $X \in \mathbb{R}^{n\times p}$ and their labels $L \in \mathbb{R}^{n\times 1}$, The joint sparse solution can be found:

$$\begin{aligned}(\alpha_1, W) = argmin_{\alpha_1,W} \ &\lambda||W\Phi_1\alpha_1 - Wy||_2 \\ &+\beta F(W, X, L) + \tau_1||\alpha_1||_1 + \tau_2||diag(W)||_1 \end{aligned} \tag{6}$$

where $F(W, X, L)$ is the loss function in the selected feature space for training dataset and samples in current frame. The τ_1 and τ_2 are the sparse parameters. As we explained before, the parameters τ_1 and τ_2 in (6) have no direct physical meaning and therefore it is difficult to tune their values. In our algorithm, we apply greedy algorithm to directly solve the original l_0 minimization problem for sparse representation. In this way (6) can be rewritten as:

$$\begin{aligned}(\alpha_1, W) = argmin_{\alpha_1,W} \ &\lambda||W\Phi_1\alpha_1 - Wy||_2 + \beta F(W, X, L), \\ \text{subject to: } &||diag(W)||_0 \leq K_0, ||\alpha||_0 \leq K_1 \text{ and } ||f||_0 \leq K_2. \end{aligned} \tag{7}$$

As it is hard to find an optimum solution for (6) when both α_1 and W are unknown, we solve (7) using two stage dynamic group sparsity optimization with greedy method. The first stage is to select the sparse set of features that

are most discriminative in separating the target from the background. Then the generative likelihood of each sample is estimated in the second stage with sparse representation. The details of the algorithm are shown in Algorithm 1. We will explain each stage in the following sections.

Feature selection. Given a set of training data $X = \{x_i \in \mathbb{R}^{1\times p}\}$ with $L = \{l_i\}, i = 1 \ldots n$ as the labels. The term $F(W, X, L)$ in equation 6 is defined as

$$F(W, X, L) = e^{-\sum_{i=1}^{n}(x_i w)l_i}, \tag{8}$$

where $w \in \mathbb{R}^{p\times 1}$ is a sparse vector. The j-th feature is selected if $w_j \neq 0$. The solution to minimize $F(W, X, L)$ can be found by solving the following sparse problem

$$w^* = argmin_w \, ||Xw - L||, \text{subject to: } ||w||_0 \leq K_0 \tag{9}$$

where K_0 is the max number of features will be selected. Here we want to emphasize that using greedy method for optimization, the parameter K_0 does have physical meaning corresponding to the number of features we plan to select. Considering Haar-like features, we do have the spatial relationship between neighborhood features. For example, if a small patch is occluded, the features extracted from this region will tend to be treated as a group in sparse optimization. Let $N_w(i, j)$ as the value of j-th neighbor of i-th feature, the support set is pruned based on Z

$$z_i = w_i^2 + \sum_{j=1}^{\tau} \theta_j^2 N_w^2(i, j), i = 1 \ldots p \tag{10}$$

in DGS taking the neighborhood relationship into consideration, where θ is the weight of neighbors. With the optimal w found by DGS, The diagonal matrix W can be constructed as

$$W_{j,j} = \begin{cases} 1, w_j^* \neq 0 \\ 0, otherwise; \end{cases} \tag{11}$$

Benefiting from the sparse solution to (9), we will be able to use advanced high dimensional features without sacrificing the efficiency of the algorithm. The other benefit is the object selection in the target region. The target templates usually contain some background features which are not linear. By doing discriminative feature selection, features from background pixels in the target templates are eliminated. The target template library is therefore more efficient and robust.

Sparse Reconstruction. After we calculate the weighting matrix W, the α and f in equation (6) can be found in the second stage

$$(\alpha, f) = argmin_{\alpha,f} \, ||W\Phi_1\alpha_1 - Wy||, \text{ subject to: } ||\alpha||_0 \leq K_1 \text{ and } ||f||_0 \leq K_2. \tag{12}$$

where $\Phi_1 = [\Phi, I]$ and $\alpha_1 = \begin{bmatrix} \alpha \\ f \end{bmatrix}$. Let $W' \in \mathbb{R}^{K_0 \times p}$ as the matrix contains all nonzero rows of W. We define $\Phi' = W'\Phi$ and $y' = W'y$. Please notify that in this step we already reduced the feature dimension from $p \times m$ to $K_0 \times m$ where m is the number of templates in the target library. In this stage the following equation is solved

$$(\alpha^*, f^*) = argmin_{\alpha,f} \left|\left| [\Phi',\ W'] \begin{bmatrix} \alpha \\ f \end{bmatrix} - y' \right|\right|, \text{ subject to: } \begin{matrix} ||\alpha||_0 \leq K_1 \\ ||f||_0 \leq K_2 \end{matrix}. \quad (13)$$

Here the sparsity parameters K_1 and K_2 have clear physical meaning, where K_1 controls the sparsity of a target template representation and K_2 controls the tolerance of occlusion. Then the likelihood of the testing sample y as target is $e^{-|\Phi'\alpha^* - y'|_2}$ and the final result is obtained by maximizing the $p(\chi_t|z_{1:t})$.

As we have already shown in Figure 1, the target templates have group structure and the temporally consecutive templates are likely to fall into the same group. The correct target sample can be reconstructed by sparse grouped templates. In our algorithm, we take into consideration the relationship between the template neighbors and tend to select grouped templates. This lead to a sparse vector in global but dense in local grouped consecutive templates. The l_1 minimization algorithm with non-negative constraints in [22] provides very sparse representation in template reconstruction coefficients, but it is vulnerable to outliers, namely, one single mistake in a template library can lead to complete tracking failure. For example, if a background sample is added into the template incorrectly, in an static background, it probably will have high matching likelihood since they are static most of time and can often find the perfect reconstruction. We avoid this problem in our algorithm by forcing a group selection of sparse coefficients. Since the outlier template is not in the same linear space as its neighbors, this can prevent it from being selected as it will lead to large reconstruction errors where even a standalone matching has a high score.

Once the tracking result is confirmed, the template library is incrementally updated as [22]. The samples with high likelihood and near the target are added to the training set as positive while the others are added as negative samples. This procedure is repeated for each frame in a whole sequence. The joint optimization of the two stage sparsity problem thus provides a fast, robust and accurate tracking result.

4 Experiments

The proposed tracking algorithm is evaluated using five challenging sequences with 3217 frames in total. The method is compared with three latest state-of-art tracking methods named L1 tracker(L1) [22], Incremental Visual Tracking (IVT) [14], Multiple Instance Learning(MIL) [4]. The tracking results of the compared algorithms are obtained by running the binaries or source code provided by their authors using the same initial positions. The source code of L1, IVT, MIL can

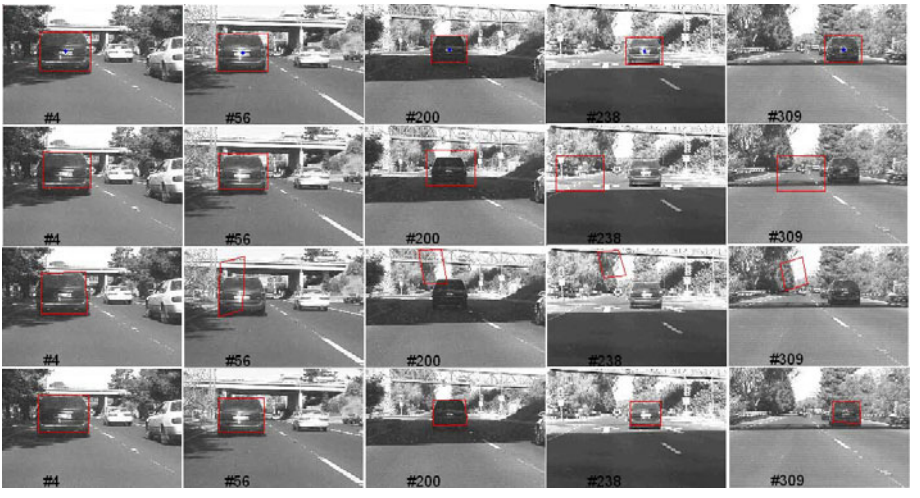

Fig. 2. The tracking results of a car sequence in an open road environment. The vehicle was driven beneath a bridge which led to large illumination changes. Results from our algorithm, MIL, L1, and IVT are given in the first, second, third, and fourth row, respectively.

be obtained from the URLs [1] [2] [3]. The first, second, third and fourth sequences were obtained from [14], and the fifth sequence was downloaded from [4].

In Section 4.1 we present the visual evaluation of the comparative tracking results. Several frames in five sequences are shown in the figures. Detailed quantitative evaluation of the comparative tracking are presented in Section 4.2. The tacking error-time curves of four sequences are plotted. Both visual and quantitative results demonstrate that our method provides more robust and accurate tracking results.

4.1 Visual Evaluation of Comparative Experiment Results

The first sequence was captured in an open road environment. The tracking results of the 4, 56, 200, 238, 309 are presented in Figure 2. The L1 starts to show some drifting on the 56-th frame. The MIL starts to show some target drifting (on the 200-th frame) and finally loses the target (the 238-th frame). IVT can track this sequence quite well. The target was successfully tracked using our proposed algorithm during the entire sequence.

The second sequence is to track a moving face. The 2, 47, 116, 173, and 222 frames are presented in Figure 4.1. The L1 algorithm fails to track the target when there are both pose and scale changes, shown in the 116-th frame. The MIL method can roughly capture the position of the object, but does have some target

[1] http://www.ist.temple.edu/ hbling/code_data.htm
[2] http://www.cs.toronto.edu/ dross/ivt/
[3] http://vision.ucsd.edu/ bbabenko/project_miltrack.shtml

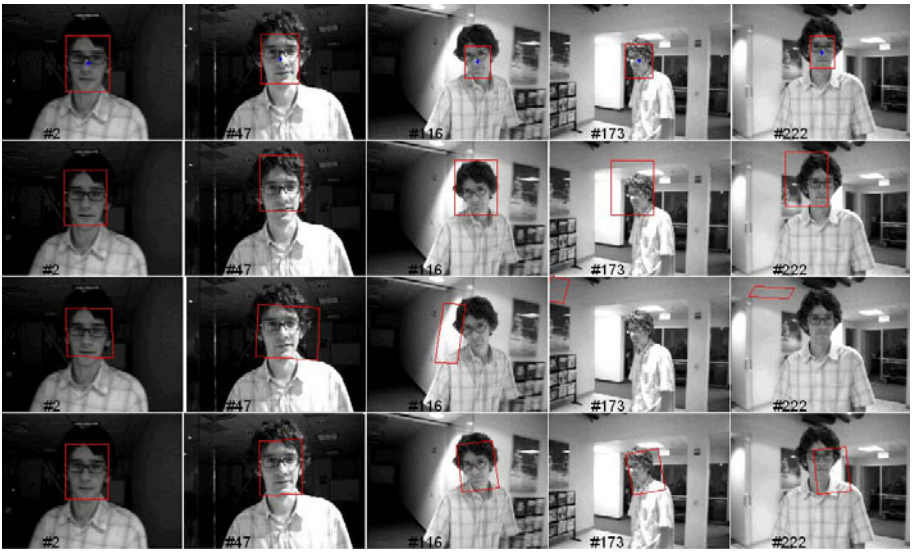

Fig. 3. The tracking results of a moving face sequence, which has large pose variation, scaling, and illumination changes. The order of the row sequences is the same as Figure 2.

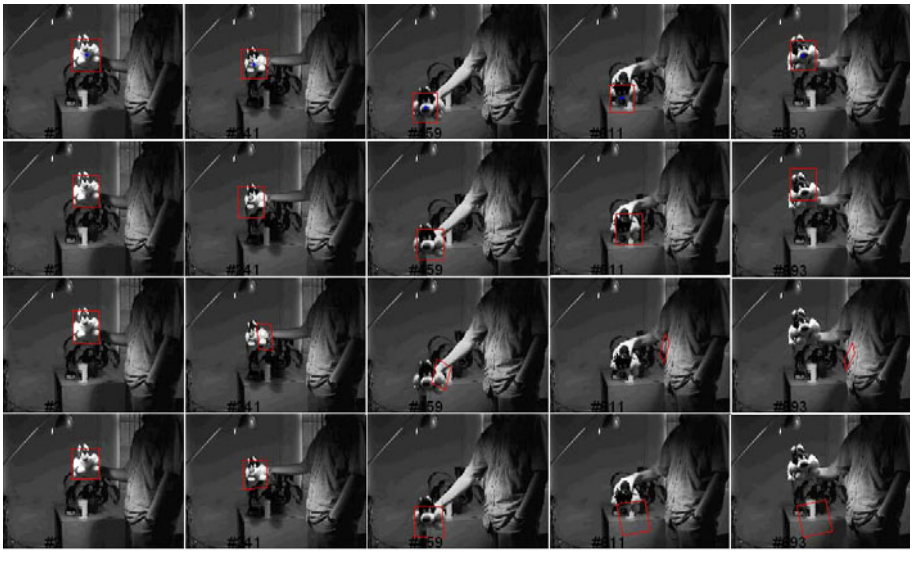

Fig. 4. The tracking results of a plush toy moving around under different pose and illumination conditions. The order of the rows is the same as in Figure 2.

drift problems, especially in the 173-th and 222-th frame. Our proposed two stage sparse tracking algorithm can track the moving face accurately through the whole sequence while the IVT produces some errors, especially on the 222-th frame.

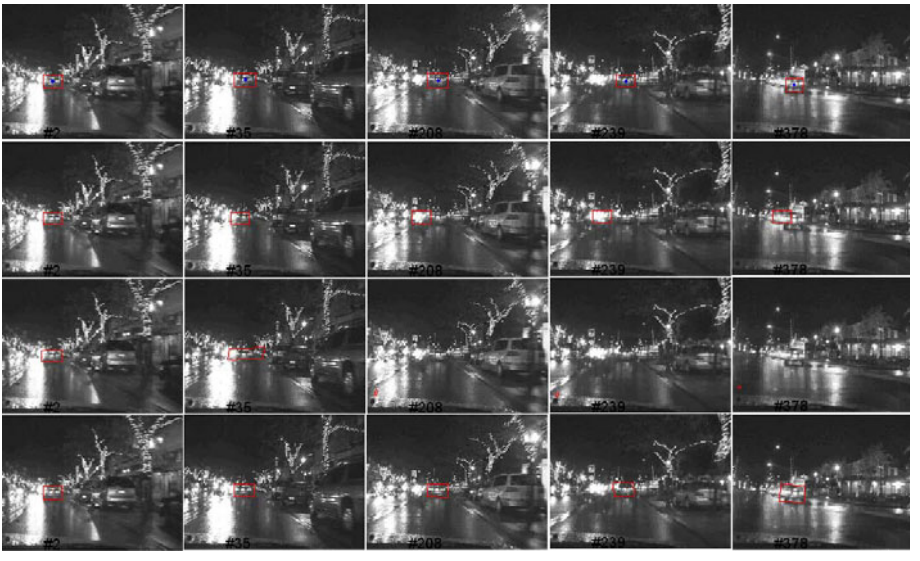

Fig. 5. The tracking results of the car sequence in a dark environment. This sequence has low resolution and poor contrast, which introduce some landmark ambiguity. The order of the row sequences is the same as Figure 5.

The third image sequence with frame 2, 241, 459, 611, and 693 is shown in Figure 6. The L1 method starts to have some drifting problem from roughly the 200-th frames, shown in the 241-th and 459-th frame. The MIL algorithm provides very good tracking results in this sequence. IVT fails to follow the object on the 611-th frame after major pose variation and can not be recovered. Our algorithm provides robust and accurate tracking result for this long sequence.

In the fourth sequence, the vehicle was driven in a very dark environment and captured from another moving vehicle. The 2, 35, 208, 239, 378 frames are presented in Figure 4.1. The L1 algorithm starts to fail to track the target from the 35-th frame. The MIL can roughly capture the position of the object before, but starts to have target drift problem from the 208-th frame distracted by light. IVT can track the target through the whole video sequence but it is not as accurate as our results, which can be found in the 378-th frame.

The results of the fifth sequence are shown in Figure 6. In this sequence we show the robustness of our algorithm in handling occlusion. The frame indexes are 10, 427, 641, 713, and 792. Starting from the 641-th frame, our method perform consistently better compared with the other methods.

4.2 Quantitative Evaluation of Comparative Experimental Results

For fair comparison, the tracking error e in each frame is measured as $e = \epsilon/d$, where ϵ is the offset of center from the ground truth and the d is the diagonal length of the target rectangle. For perfect tracking, the e should be equal to zero

Fig. 6. The tracking results of a face sequence, which includes a lot of pose variations, partial or full occlusions. The order of the row sequences is the same as Figure 2.

for each frame. In Table 1, we compared the quantitative e using our proposed algorithm with L1, MIL and IVT.

The best result in each column is shown in bold in Table 1. The missing column represents the number of frames where the $e > 1$. For a fair comparison, we do not count these failing frames when computing the *overall mean and variance* in the 7-th and the 8-th columns in Table 1. Measured by the public open benchmark, on average our algorithm only has 7% of drifting errors and never misses one single frame in the five tracking sequences which contain thousands of frames in total. In Figure 7 we present the tracking error-time curve. We can see that except for the fifth sequence, in which we obtain similar results as IVT (IVT will intend to shrink the window to very small size but won't lose the center of the target, as shown in Figure 6), our algorithm does outperform the other methods. The method is computationally efficient. Even using a MATLAB implementation, it can process two frames/second.

Table 1. The overall quantitative tracking performance comparison of proposed robust tracking method with two stage sparse optimization, L1 [22], MIL [4], and IVT [14].

	Mean					Overall				
	Seq1	Seq2	Seq3	Seq4	Seq5	Mean	Variance	Median	Max	Missing
L1	1.10	1.31	0.89	3.22	0.21	0.38	0.26	1.34	5.09	828
MIL	1.02	0.34	0.17	1.16	0.12	0.31	0.29	0.46	3.82	55
IVT	0.04	0.09	1.15	**0.07**	0.13	0.08	0.08	0.05	5.82	470
Proposed Method	**0.03**	**0.08**	**0.16**	0.08	**0.12**	**0.07**	**0.06**	**0.04**	**0.34**	**0**

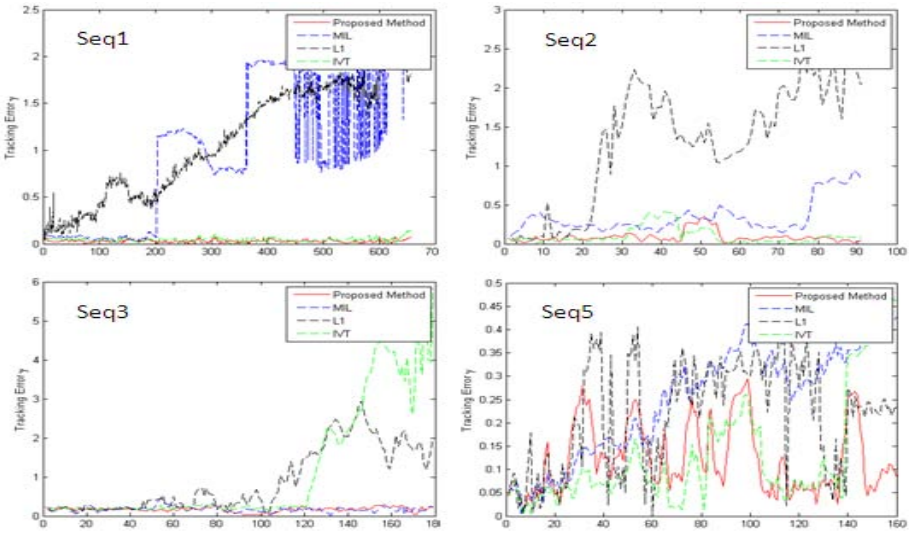

Fig. 7. The tracking accuracy e for each frame in four different sequences

5 Conclusion

We have proposed an online robust and fast tracking algorithm using a two stage sparse optimization approach. No shape or motion priors are required for this algorithm. Both the training set and the template library models are online updated. Two stage sparse optimization is solved jointly by minimizing the target reconstruction error and maximizing the discriminative power by selecting a sparse set of features. The experimental results demonstrate the effectiveness of our method in handling a number of challenging sequences.

Acknowledgement

This research is supported, in part, by UMDNJ Foundation Funding #66-09.

References

1. Yang, M., Wu, Y., Hua, G.: Context-aware visual tracking. PAMI 31, 1195–1209 (2009)
2. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. ACM Comput. Surv. 38, 13–32 (2006)
3. Avidan, S.: Ensemble tracking. PAMI 29, 261–271 (2007)
4. Babenko, B., Yang, M.H., Belongie, S.: Visual tracking with online multiple instance learning. In: CVPR (2009)
5. Breitenstein, M.D., Reichlin, F., Leibe, B., Koller-Meier, E., Gool, L.V.: Robust tracking-by-detection using a detector confidence particle filter. In: ICCV (2009)
6. Hess, R., Fern, A.: Discriminatively trained particle filters for complex multi-object tracking. In: CVPR (2009)
7. Grabner, H., Bischof, H.: On-line boosting and vision. In: CVPR, vol. 1, pp. 260–267 (2006)

8. Grabner, H., Grabner, M., Bischof, H.: Real-time tracking via on-line boosting. In: British Machine Vision Conference, vol. 1, pp. 47–55 (2006)
9. Matthews, I., Baker, S.: Active appearance models revisited. IJCV 60, 135–164 (2004)
10. Black, M.J., Jepson, A.D.: Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. IJCV 26, 329–342 (1998)
11. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. PAMI 24, 603–619 (2002)
12. Matthews, L., Ishikawa, T., Baker, S.: The template update problem. PAMI 26, 810–815 (2004)
13. Porikli, F., Tuzel, O., Meer, P.: Covariance tracking using model update based on Lie algebra. In: CVPR, vol. 1, pp. 728–735 (2006)
14. Ross, D., Lim, J., Lin, R.S., Yang, M.H.: Incremental learning for robust visual tracking. IJCV 77, 125–141 (2008)
15. Xue, M., Zhou, S.K., Porikli, F.: Probabilistic visual tracking via robust template matching and incremental subspace update. In: IEEE International Conference on Multimedia and Expo, pp. 1818–1821 (2007)
16. Zhou, S.K., Chellappa, R., Moghaddam, B.: Visual tracking and recognition using appearance-adaptive models in particle filters. ITIP 13, 1491–1506 (2004)
17. Yang, L., Georgescu, B., Zheng, Y., Meer, P., Comaniciu, D.: 3D ultrasound tracking of the left ventricle using one-step forward prediction and data fusion of collaborative trackers. In: CVPR (2008)
18. Gu, J., Nayar, S.K., Grinspun, E., Belhumeur, P.N., Ramamoorthi, R.: Compressive structured light for recovering inhomogeneous participating media. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 845–858. Springer, Heidelberg (2008)
19. Cevher, V., Sankaranarayanan, A., Duarte, M.F., Reddy, D., Baraniuk, R.G., Chellappa, R.: Compressive sensing for background subtraction. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 155–168. Springer, Heidelberg (2008)
20. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Discriminative learned dictionaries for local image analysis. In: CVPR (2008)
21. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. PAMI 31, 210–227 (2009)
22. Mei, X., Ling, H.: Robust visual tracking using l_1 minimization. In: ICCV (2009)
23. Donoho, D.: Compressed sensing. IEEE Transactions on Information Theory 52, 1289–1306 (2006)
24. Huang, J., Huang, X., Metaxas, D.: Learning with dynamic group sparsity. In: ICCV (2009)
25. Grabner, H., Leistner, C., Bischof, H.: Semi-supervised on-line boosting for robust tracking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 234–247. Springer, Heidelberg (2008)
26. Yu, Q., Dinh, T.B., Medioni, G.: Online tracking and reacquisition using co-trained generative and discriminative trackers. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 678–691. Springer, Heidelberg (2008)
27. Mallat, S., Zhang, Z.: Matching pursuits with timefrequency dictionaries. IEEE Transactions on Signal Processing 41, 3397–3415 (1993)
28. Tropp, J., Gilbert, A.: Signal recovery from random measurements via orthogonal matching pursuit. IEEE Transactions on Information Theory 53, 4655–4666 (2007)
29. Dai, W., Milenkovic, O.: Subspace pursuit for compressive sensing: Closing the gap between performance and complexity. CoRR abs/0803.0811 (2008)
30. Needell, D., Tropp, J.: Cosamp: Iterative signal recovery from incomplete and inaccurate samples. Applied and Computational Harmonic Analysis (2008)