# Tracking via Robust Multi-Task Multi-View Joint Sparse Representation

Zhibin Hong[1], Xue Mei[2], Danil Prokhorov[2], and Dacheng Tao[1]

[1]Centre for Quantum Computation and Intelligent Systems, Faculty of Engineering and Information Technology, University of Technology, Sydney, NSW 2007, Australia

[2] Toyota Research Institute, North America, Ann Arbor, MI 48105, USA

{zhibin.hong@student.,dacheng.tao@}uts.edu.au, {xue.mei, danil.prokhorov}@tema.toyota.com

## Abstract

*Combining multiple observation views has proven beneficial for tracking. In this paper, we cast tracking as a novel multi-task multi-view sparse learning problem and exploit the cues from multiple views including various types of visual features, such as intensity, color, and edge, where each feature observation can be sparsely represented by a linear combination of atoms from an adaptive feature dictionary. The proposed method is integrated in a particle filter framework where every view in each particle is regarded as an individual task. We jointly consider the underlying relationship between tasks across different views and different particles, and tackle it in a unified robust multi-task formulation. In addition, to capture the frequently emerging outlier tasks, we decompose the representation matrix to two collaborative components which enable a more robust and accurate approximation. We show that the proposed formulation can be efficiently solved using the Accelerated Proximal Gradient method with a small number of closed-form updates. The presented tracker is implemented using four types of features and is tested on numerous benchmark video sequences. Both the qualitative and quantitative results demonstrate the superior performance of the proposed approach compared to several state-of-the-art trackers.*

## 1. Introduction

Tracking problems can involve data that is represented by multiple views [1] of various types of visual features including intensity [28], color [4], edge [14], wavelet [12] and texture. Exploiting these multiple sources of information can significantly improve tracking performance as a result of their complementary characteristics [2][14][7][18]. Given these cues from multiple views, an important problem is how to integrate them and build an appropriate model to explore their mutual dependencies and independencies.

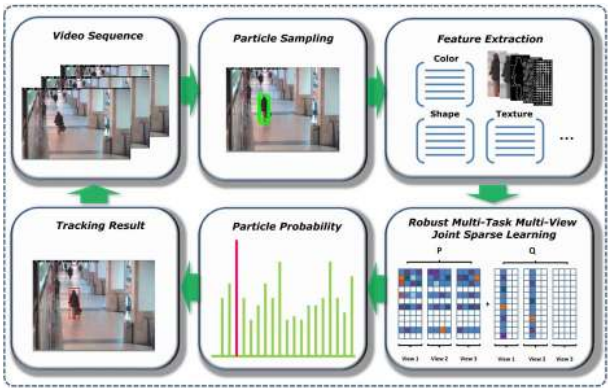


Figure 1. A flowchart to illustrate the proposed tracking framework.

Sparse representation has recently been introduced for tracking [19], in which a tracking candidate is sparsely represented as a linear combination of target templates and trivial templates. In particle filter-based tracking methods, particles around the current state of the target are randomly sampled according to a zero-mean Gaussian distribution. Each particle shares dependencies with other particles. Multi-task learning aims to improve the performance of multiple related tasks by exploiting the intrinsic relationship among them. In [35], learning the representation of each particle is viewed as an individual task and a multi-task learning with joint sparsity for all particles is employed. However, they assume that all tasks share a common set of features, which generally does not hold in visual tracking applications, since outlier tasks often exist. For example, a small number of particles sampled far away from the majority of particles may have little overlap with other particles and will be considered as outliers. In addition, [19], [35] only use the intensity feature to model the appearance change of the target. The intensity appearance model with $\ell_1$ minimization is very robust to partial occlusion, noise, and other tracking challenges [19]. However, it is very

[1]Regarding the term multi-view learning [25] [29], we follow the machine learning convention, in which views refer to different feature subsets used to represent particular characteristics of an object.

sensitive to shape deformation of targets such as non-rigid objects.

To overcome the above problems, we propose to employ other visual features such as color, edge, and texture to complement intensity in the appearance representation, and to combine a multi-view representation with a robust multi-task learning [9] to solve the visual tracking problem (Figure 1). Within the proposed scheme, the sparse representation for each view is learned as a linear combination of atoms from an adaptive feature dictionary, i.e. each view has its own sparse representation instead of sharing an identical one, which enables the tracker to capture different statistics carried by different views. To exploit the interdependencies shared between different views and particles, we impose the $\ell_{1,2}$-norm group-sparsity regularization on the representation matrix to learn the multi-view sparse representation jointly in a multi-task manner. To handle the outlier particles from particle sampling, we decompose the sparse representation into two collaborative parts, thereby enabling them to learn representative coefficients and detect outlier tasks simultaneously. An efficient Accelerated Proximal Gradient (APG) [22] scheme is employed to obtain the optimal solution via a sequence of closed-form updates.

Our contribution is four-fold: 1) we utilize multiple types of features in a sparse representation-based framework for tracking. Compared to previous related trackers [15] [19][35], the new tracker is not only able to take advantage of the robustness to occlusion from sparse representation, but also introduces complementary multiple-view representation for robust appearance modeling; 2) we treat every view in each particle as an individual task and jointly consider the underlying relationship shared among different views and different particles in a multi-task learning framework; 3) to capture the outlier tasks that frequently emerge in the particle sampling process, we employ a robust multi-task scheme by decomposing the coefficient matrix into two collaborative components; and 4) outlier rejection helps identify outlier tasks and improves resampling efficiency by setting posterior probabilities of outliers to zero and making sure they are not sampled in the resampling process.

## 2. Related Work

An extensive review on tracking and multi-view learning is beyond the scope of this paper. We refer readers to a comprehensive survey [31] for more details about existing trackers, and an extensive survey on multi-view learning can be found in [30]. In this section, we review the works related to our method including popular single-view based trackers, multi-view based trackers and multi-task learning.

Numerous existing trackers only use single feature and solve tracking in various ways. For instance, Comaniciu *et al*. [5] introduce a spatial kernel to regularize the color histogram-based feature representation of the target, which enables tracking to be reformulated as a gradient-based optimization problem solved by mean-shift. Babenko *et al*. [1] employ Multiple Instance Learning (MIL) equipped with a Haar feature pool to overcome the label ambiguity problem. In [26], Ross *et al*. present a tracking method that incrementally learns a low-dimensional subspace representation based on intensity features. Recently, Kalal *et al*. [13] propose a new tracking paradigm that combines the classical Lucas-Kanade method-based tracker with an online learned random-forest based detector using pixel-wise comparison features. The learned detector is notable for enabling reacquisition following tracking failures.

The above trackers nevertheless tend to be vulnerable in particular scenarios due to the limitations of the adopted features. Various methods aim to overcome this problem by taking advantage of multiple types of features to enable a more robust tracker [21][14][32]. In [21], Moreno-Noguer *et al*. propose a probabilistic framework allowing the integration of multiple features for tracking by considering cue dependencies. Kwon and Lee [14] propose Visual Tracking Decomposition (VTD) that employs Sparse Principal Component Analysis (SPCA) to construct multiple basic observation models (basic trackers) based on multiple types of features. An Interactive Markov Chain Monte Carlo (IMCMC) scheme is then used to integrate all the basic trackers.

Sparse representation was recently introduced for tracking in [19] which casts tracking as a sparse representation problem in a particle filter framework [11] which was later exploited in [15][16][20]. In [35], a multi-task learning [3] approach is applied to tracking by learning a joint sparse representation of all the particles in a particle filter framework. Compared to the original L1 tracker [19] that pursues the sparse representation independently, Multi-Task Tracking (MTT) achieves more robust performance by exploiting the interdependency between particles. Multi-task sparse learning has also been successfully applied to image classification [33], in which a multi-task joint covariate selection model is used to classify a query image using multiple features from a set of training images, and a class-level joint sparsity regularization is imposed on class-level representation coefficients.

Motivated by the above advances, in this paper, we propose a Multi-Task Multi-View Tracking (MTMVT) method based on joint sparse representation to exploit the related information shared between particles and views in order to obtain improved performance.

## 3. Multi-task Multi-view Sparse Tracker

The L1 tracker [19] tackles tracking as finding a sparse representation in the template subspace. The representation is then used in a particle filter framework for visual tracking. However, appearance representation based only

on intensity is prone to failure in difficult scenarios such as tracking non-rigid objects. Employing multiple types of features has proven to be beneficial for tracking because the ensemble of multiple views provides a comprehensive representation of the target appearance undergoing various changes such as illumination and deformation. However, combining multiple views by simply concatenating features into a high-dimensional feature vector is inappropriate, since different features have different statistical properties. Inspired by previous works [33][35], the dependencies of these views as well as the intrinsic relationship of sampled particles should be jointly considered. In this section, we propose to employ other visual features such as color, edge, and texture to complement intensity in the target appearance representation, and to combine a multi-view representation with a robust multi-task learning [9] to solve the visual tracking problem.

We denote $y_t$ as the state variable describing the location and shape of a target at time frame $t$. The tracking problem can then be formulated as an estimation of the state probability $p(y_t|x_{1:t})$, where $x_{1:t} = \{x_1, \ldots, x_t\}$ represents the observations from previous $t$ frames. To model the observation likelihood $p(x_t|y_t)$, a region corresponding to state $y_t$ is first cropped from the current frame. Multiple features are then extracted from the region and used to form a 1D feature vector $x_t$.

### 3.1. Sparse Representation-based Tracker

In [19], the sparse representation of intensity feature $x$ is formulated as the minimum error reconstruction through a regularized $\ell_1$ minimization problem with nonnegativity constraints

$$\min_{w} \parallel Mw - x \parallel_2^2 + \lambda \parallel w \parallel_1 \quad , \quad \text{s.t. } w \succcurlyeq 0 \ , \qquad (1)$$

where $M = \left[D, I, -I\right]$ is an over-complete dictionary that is composed of target template set $D$ and positive and negative trivial template sets $I$ and $-I$. Each column in $D$ is a target template generated by reshaping pixels of a candidate region into a column vector; and each column in the trivial template sets is a unit vector that has only one nonzero element. $w = \left[a^\top, e^{+\top}, e^{-\top}\right]^\top$ is composed of target coefficients $a$ and positive and negative trivial coefficients $e^+$, $e^-$ respectively.

Finally, the observation likelihood is derived from the reconstruction error of $x$ as

$$p(x|y) = \frac{1}{\Gamma} \exp\{-\alpha \parallel Da - x \parallel^2\} \ , \qquad (2)$$

where $a$ is obtained by solving the $\ell_1$ minimization (1), $\alpha$ is a constant controlling the shape of the Gaussian kernel, and $\Gamma$ is a normalization factor.

### 3.2. Robust Multi-task Multi-view Sparse Learning

We consider $n$ particle samples, each of which has $K$ different views (e.g., color, shape and texture). For each

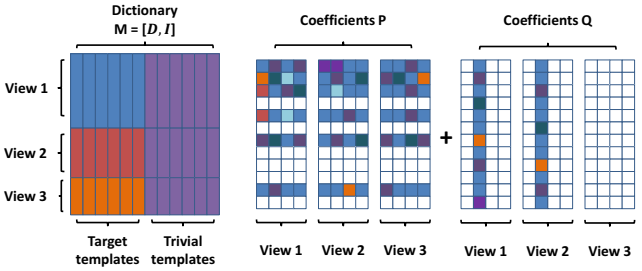


Figure 2. The illustration for the structure of the learned coefficient matrices $P$ and $Q$, where entries of different color represent different learned values, and the white entries in $P$ and $Q$ indicate the zero rows and columns. Note that this figure demonstrates a case that includes four particles, where the second particle is an outlier whose coefficients in $Q$ comprise large values.

view index $k = 1, \ldots, K$, denote $X^k \in \mathbb{R}^{d_k \times n}$ as the feature matrix which is a stack of $n$ columns of normalized particle image feature vectors of dimension $d_k$, where $d_k$ is the dimension for the $k$th view. We denote $D^k \in \mathbb{R}^{d_k \times N}$ as the target dictionary in which each column is a target template from the $k$th view, where $N$ is the number of target templates. The target dictionary is combined with trivial templates $I_{d_k}$ to construct the complete dictionary $M^k = [D^k, I_{d_k}]$.

Based on the fact that most of the particles are relevant and outliers often exist, we introduce a robust multi-task learning scheme [9] to capture the underlying relationships shared by all tasks. We jointly evaluate $K$ feature view matrices $\{X^1, \ldots, X^K\}$ with $n$ particles and learn the latent representations $\{W^1, \ldots, W^K\}$. The decomposed matrices $W^k$s enable different views of particles to have different learned representations, and therefore exploit the independency of each view and capture the different statistical properties. Moreover, each representation matrix $W^k$ is constructed by two collaborative components $P^k$ and $Q^k$, where $P^k$ is regularized by row sparse constraint, which assumes that all particles share the same basis, while $Q^k$ is regularized by column sparse constraint, which enables the capture of outlier tasks.

The same columns from each view in the dictionary should be activated to represent the particle in a joint sparse manner, since the corresponding columns represent the same sample of the object. Therefore, the corresponding decomposed weight matrices $P^k$s and $Q^k$s from all the views can be stacked horizontally to form two bigger matrices $P$ and $Q$, respectively. Each of them consists of the coefficients across all the views. Group lasso penalty $\ell_{1,2}$ is applied to row groups of the first component $P$ for capturing the shared features among all tasks over all views, where we define $\|P\|_{1,2} = \sum_i (\sum_j P_{i,j}^2)^{1/2}$, and $P_{i,j}$ denotes the entry in the $i$th row and $j$th column in the matrix $P$. The same group lasso penalty is imposed on column groups of the second component $Q$ to identify the outlier tasks simul-

taneously. The multi-view sparse representations for all particles can be obtained from the following problem

$$\min_{W,P,Q} \frac{1}{2}\sum_{k=1}^{K} ||M^k W^k - X^k||_F^2 + \lambda_1||P||_{1,2} + \lambda_2||Q^\top||_{1,2} \,, \tag{3}$$

where $W^k = P^k + Q^k, P = [P^1, \ldots, P^K]$, $Q = [Q^1, \ldots, Q^K]$, and $\lambda_1$ and $\lambda_2$ are the parameters controlling the sparsity of $P$ and $Q$, respectively. Figure 2 illustrates the structure of the learned matrices $P$ and $Q$.

Note that the stacking of $P^k$s and $Q^k$s requires that $M^k$s have the same number of columns. However, we can pad the matrices $M^k$s with zero columns to make them the same number of columns in order to apply (3). The coefficients associated with the zero columns will be zeros based on the sparsity constraints from $\ell_1$ regularization and do not impact the minimization function in terms of the solution. Without loss of generality, we assume $M^k$s are sorted in descending order of the number of columns $n_k$, that is, $n_1 \geq n_2 \geq \ldots \geq n_K$. The new $\hat{M}^k$ is defined as the zero padded matrix of $M^k$, that is, $\hat{M}^k = [M^k, 0^k]$, where $0^k \in \mathbb{R}^{d_k \times (n_1 - n_k)}$ and every element in $0^k$ is zero. We can replace $M^k$ in (3) with $\hat{M}^k$ and solve the same minimization problem. For a more intuitive view of the proposed formulation, we visualize an empirical example of the learned sparse coefficients in Figure 4, where $W = [A^\top, E^\top]^\top$ consists of target coefficients $A$ and trivial coefficients $E$ respectively.

In reference to the tracking result, the observation likelihood of the tracking candidate $i$ is defined as

$$p_i = \frac{1}{\Gamma}\exp\{-\alpha \sum_{k=1}^{K} \parallel D^k A_i^k - X_i^k \parallel^2\} \,, \tag{4}$$

where $A_i^k$ is the coefficients of the $i$th candidate corresponding to the target templates of the $k$th view. The tracking result is the particle that has the maximum observation likelihood. To handle appearance variations, the target dictionary $D$ is progressively updated similar to [19], and the templates are weighted in the course of tracking.

### 3.3. Outlier Rejection

Although a majority of particles will share the same dictionary basis, some outlier tasks may exist. These are the particles sampled far away from the target that have little overlap with other particles. The proposed MTMVT in (3) is capable of capturing the outlier tasks by introducing the coefficient matrix $Q$. In particular, if the sum of the $\ell_1$ norm of the coefficients for the corresponding $i$th particle is larger than an adaptive threshold $\gamma$, as

$$\sum_{k=1}^{K} \mid Q_i^k \mid > \gamma \,, \tag{5}$$

where $Q_i^k$ is the $i$th column of $Q^k$, then it will be identified as an outlier and its observation likelihood will be set to zero, and thus the outliers will be ignored in the particle resampling process. Therefore, we utilize samples more efficiently without wasting samples on the outliers. By denoting the number of detected outlier tasks as $n_o$, the threshold $\gamma$ is updated as follows

$$\begin{cases} \gamma_{new} = \gamma_{old}\kappa, \; n_o > N_o \\ \gamma_{new} = \gamma_{old}/\kappa, \; n_o = 0 \\ \gamma_{new} = \gamma_{old}, \; 0 < n_o \leq N_o \,, \end{cases} \tag{6}$$

where $\kappa$ is a scaling factor, and $N_o$ is a predefined threshold for the number of outliers. We select $\gamma = 1$, $\kappa = 1.2$ and $N_o = 20$ based on experiments. Figure 3 illustrates examples showing detected outliers.

Figure 3. Examples of detected outlier tasks. The green bounding boxes denote the outlier particles and the red bounding box denotes the tracked target. The outliers are detected out of 400 sampled particles. There are two outliers in the left frame and six outliers in the right frame.

### 3.4. Optimization Algorithm

This section shows how to solve (3) efficiently. Note that the objective function in (3) is a composite function of two parts, a differential empirical loss function $\ell(P, Q)$ and a convex non-smooth regularization $r(P, Q)$, which has been extensively studied [9][22][3]. The Accelerated Proximal Gradient (APG) method [3] is employed because of its well-known efficiency. In contrast to traditional subgradient-based methods that converge at sublinear rate, APG can obtain the globally optimal solution at quadratic convergence rate, which means APG achieves $O(1/m^2)$ residual from the optimal solution after $m$ iterations.

Denote

$$\ell(P, Q) = \frac{1}{2}\sum_{k=1}^{K} ||M^k W^k - X^k||_F^2 \,, \tag{7}$$

$$r(P, Q) = \lambda_1||P||_{1,2} + \lambda_2||Q^\top||_{1,2} \,. \tag{8}$$

We can apply the *composite gradient mapping* [22] to (3) and construct the following function

$$\begin{aligned} \Phi(P, Q; R, S) =& \ell(R, S) + \langle \nabla_R \ell(R, S), P - R \rangle \\ &+ \langle \nabla_S \ell(R, S), Q - S \rangle + \frac{\eta}{2}\|P - R\|_F^2 \\ &+ \frac{\eta}{2}\|Q - S\|_F^2 + r(P, Q) \,. \end{aligned} \tag{9}$$

In $\Phi(P, Q; R, S)$ comprises the regularization term $r(P, Q)$ and the approximation of $\ell(P, Q)$ by the first order Taylor

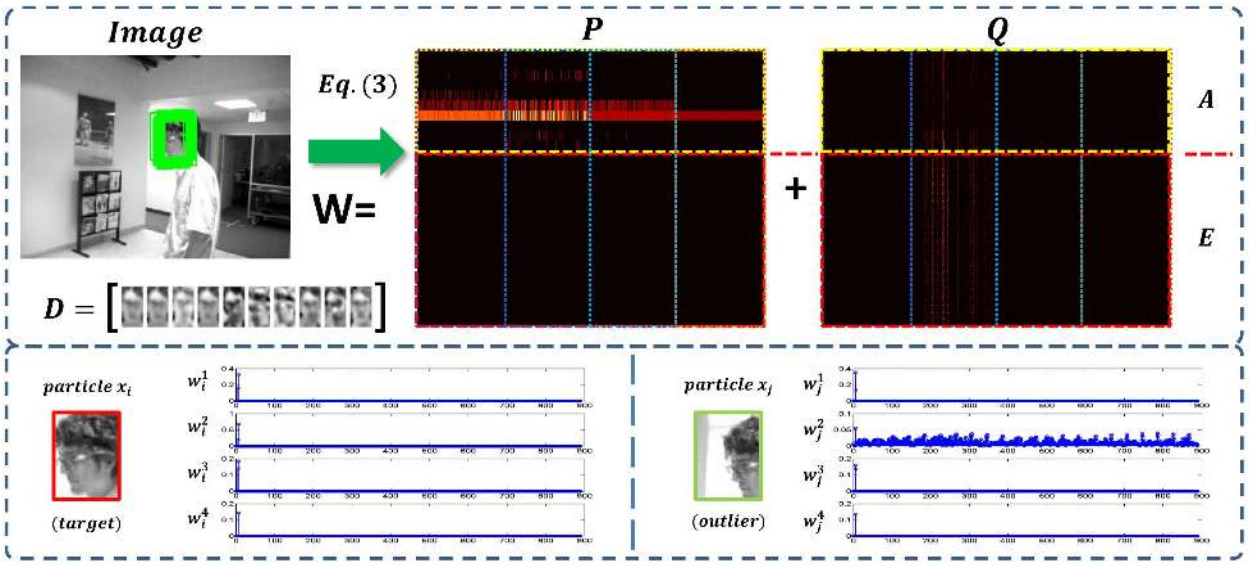


Figure 4. An example of the learned coefficients. In the top figure, we visualize the learned coefficient matrices $P$ and $Q$ for all particles across all views, which are color histograms, intensity, HOG and LBP, respectively. Each matrix consists of four column parts corresponding to four different views, where the brighter color represents a larger value in the corresponding entry. The seventh template in the dictionary is the most representative and results in brighter values in the seventh row of $P$ across all views, while some columns in Q have brighter values which indicate the presence of outliers. The bottom figures illustrate the coefficients of two particles.

expansion at point $(R, S)$ regularized as the squared Euclidean distance between $(P, Q)$ and $(R, S)$, where $\nabla_R \ell(R, S)$ and $\nabla_S \ell(R, S)$ denote the partial derivatives of $\ell(R, S)$ with respect to $R$ and $S$, and $\eta$ is a parameter controlling the step penalty.

In the $m$th APG iteration, $(R^{m+1}, S^{m+1})$ is computed as a linear combination of $(P^m, Q^m)$ and $(P^{m-1}, Q^{m-1})$, so $(R^{m+1}, S^{m+1})$ stores the historical aggregation of $(P, Q)$ in the previous iterations, which is conventionally called *aggregation step*. As suggested in [3], we set

$$\begin{aligned} R^{m+1} &= P^m + \alpha_m(\frac{1-\alpha_{m-1}}{\alpha_{m-1}})(P^m - P^{m-1}) , \\ S^{m+1} &= Q^m + \alpha_m(\frac{1-\alpha_{m-1}}{\alpha_{m-1}})(Q^m - Q^{m-1}) , \end{aligned} \tag{10}$$

where $\alpha_m$ can be set to $\alpha_0 = 1$ for $m = 0$ and $\alpha_m = \frac{2}{m+3}$ for $m \geq 1$, and $P^0$, $Q^0$, $R^1$ and $S^1$ are all set to zero matrix for the initialization. Once given the aggregation $(R^m, S^m)$, the solution for the $m$th iteration is obtained by computing the following *proximal operator*

$$(P^m, Q^m) = \arg_{P,Q} \min \Phi(P, Q; R^m, S^m) . \tag{11}$$

With simple manipulations, the optimization problem (11) can be decomposed into two subproblems for $P$ and $Q$ respectively, as

$$P^m = \arg\min_P \frac{1}{2}\|P - U^m\|_F^2 + \frac{\lambda_1}{\eta}\|P\|_{1,2} , \tag{12}$$

$$Q^m = \arg\min_Q \frac{1}{2}\|Q - V^m\|_F^2 + \frac{\lambda_2}{\eta}\|Q^\top\|_{1,2} , \tag{13}$$

where $U^m = R^m - \frac{1}{\eta}\nabla_R \ell(R^m, S^m)$ and $V^m = S^m - \frac{1}{\eta}\nabla_S \ell(R^m, S^m)$.

Following the decomposition, an efficient closed-form solution can be attained respectively for each row of $P^m$ and each column of $Q^m$ in the above subproblems (12) and (13) according to [17],

$$\begin{aligned} P^m_{i,\cdot} &= \max(0, 1 - \frac{\lambda_1}{\eta\|U^m_{i,\cdot}\|})U^m_{i,\cdot} \\ Q^m_{\cdot,i} &= \max(0, 1 - \frac{\lambda_2}{\eta\|V^m_{\cdot,i}\|})V^m_{\cdot,i} , \end{aligned} \tag{14}$$

where $P^m_{i,\cdot}$ denotes the $i$th row of $P^m$ and $Q^m_{\cdot,i}$ denotes the $i$th column of $Q^m$. Finally, the solution of (3) can be obtained by iteratively computing (14) and updating ($U^m$, $V^m$) until the convergence of ($P$, $Q$). The procedure of the presented algorithm is summarized in the *supplementary material*.

## 4. Experiments

To evaluate the effectiveness of the new tracker, it was implemented using four complementary features. We extensively validated it on twelve publicly available challenging sequences[2]. All images are resized to $320 \times 240$ as in our previous work [10]. We compared MTMVT

[2]http://vision.ucsd.edu/~bbabenko/project_miltrack.shtml; http://www.cs.toronto.edu/~dross/ivt/; http://cv.snu.ac.kr/research/~vtd/; http://www4.comp.polyu.edu.hk/~cslzhang/CT/CT.htm[34]; http://www.eng.tau.ac.il/~oron/LOT/LOT.html[24]; http://lrs.icg.tugraz.at/research/houghtrack/ [8]

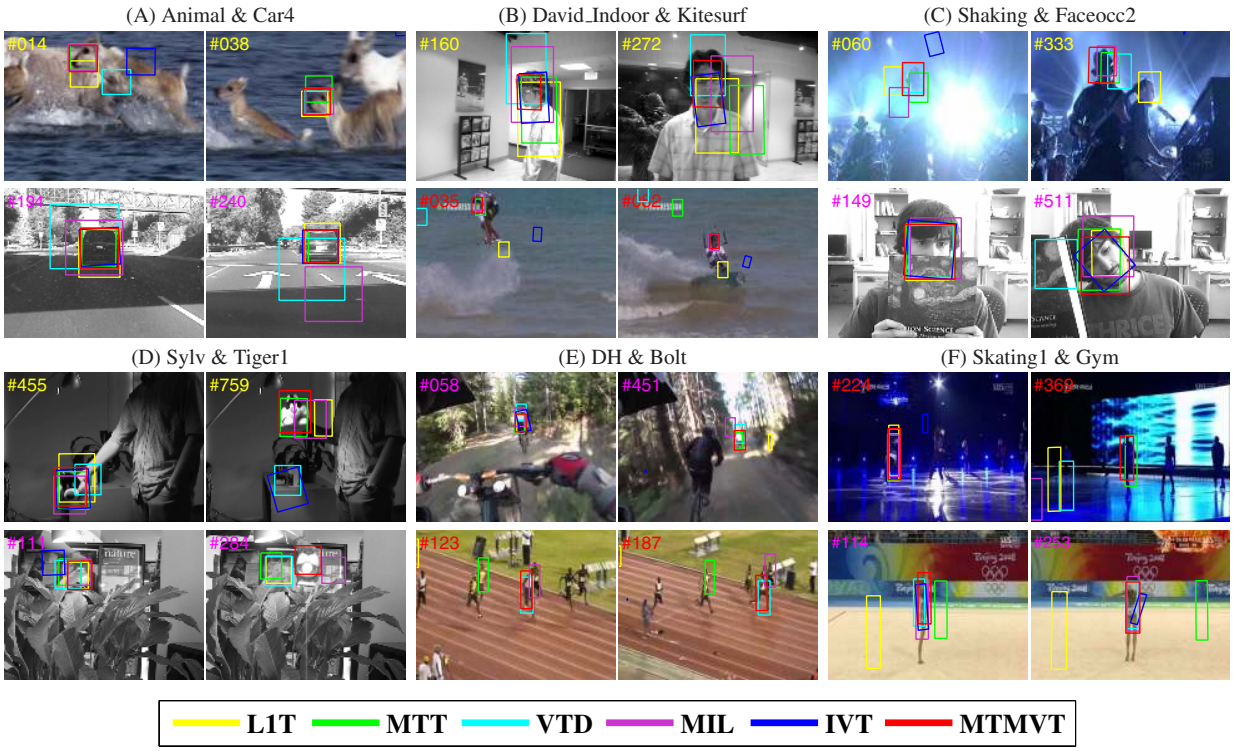


Figure 5. Tracking results of different algorithms. Frame indexes are shown in the top left of each figure.

with five other popular trackers: L1 Tracker (L1T) [19], Multi-Task Tracking (MTT) [35], tracking with Multiple Instance Learning (MIL) [1], Incremental Learning for Visual Tracking (IVT) [26], and Visual Tracking Decomposition (VTD) [14]. It should be noted that VTD is a multi-view tracker which employs hue, saturation, intensity, and edge template for the features. We conducted the experiments by running source codes provided by the original authors. The recommended parameters are set for initialization.

### 4.1. Implementation Details

To take the advantage of complementary features, we employed four popular features: color histograms, intensity, histograms of oriented gradients (HOG) [6] and local binary patterns (LBP) [23]. HOG is a gradient-based feature that captures edge distribution of an object. Local binary patterns (LBP) is powerful for representing object texture. Moreover, to ensure the quality of extracted features, a simple but effective illumination normalization method used in [27] is applied before the feature extraction. The unit-norm normalization is applied to the extracted feature vector of each particle view respectively as done in [19].

For all reported experiments, we set $\lambda_1 = \lambda_2 = 0.5$, the number of particles $n = 400$ (the same for L1T and MTT), the number of template samples $N = 10$. The template of intensity is set to one third size of the initial target (half size for those whose shorter side is less than 20), while the color histograms, HOG, LBP are extracted in a larger region that doubles the size of the intensity template.

### 4.2. Qualitative Comparison

The *Animal* and *Car4* sequences shown in Figure 5(A) track the head of a fast running deer and a moving car, respectively. The main challenges of these two sequences are the fast motion, background clutter, scale changes and illumination changes. For the *Animal* sequence, only MIL and MTMVT succeed in tracking the target over the whole sequence, while MTT is able to track most of the frames. IVT gradually drifts from the target after the second frame and totally loses the target in the seventh frame. L1T fails in the presence of fast motion and motion blur. The multi-task manner appears to make MTT and MTMVT more robust than L1T. However, MTT is not as robust as MTMVT since MTMVT takes advantage of the complementary features and is capable of detecting outlier tasks. In the *Car4* sequence, both MTMVT and IVT perfectly track the moving car despite the dramatic illumination and scale changes, which are shown in the second row of Figure 5(A). By contrast, VTD and MIL lose the target and L1T tends to include much of the background area into the bounding box when undergoing significant illumination changes.

The *David_Indoor*, *Kitesurf*, *Shaking*, *Face2* sequences track human faces under different circumstances and chal-

lenges. The experimental results show that MTMVT is able to handle the scale changes, pose changes, fast motion, occlusion, appearance variation, and angle variation problems encountered in face tracking tasks. For example, the *Shaking* sequence captures a person performing on stage. The task is to track his face under significant illumination changes and appearance variations. Our tracker is more robust to the illumination changes as a result of the employment of rich feature types. In the *David_Indoor* sequence, a moving face is tracked, which presents many challenges such as pose and scale changes. Compared to L1T and MTT, MTMVT perfectly tracks the target under different challenges due to the robustness of the additional features. From the experiments, we find that IVT is vulnerable to the appearance variations, while VTD is prone to drift in occlusion scenarios. See Figure 5(B - C).

In *Sylv* and *Tiger1* sequences, the tasks are to track moving dolls in indoor scenes. Almost all the trackers compared can track the doll in the earlier part of the *Sylv* sequence. However, L1T, MIL VTD and IVT lose the target when it undergoes pose changes. The *Tiger1* sequence is much harder due to the significant appearance changes, occlusion, and distracting background, so all trackers continuously lock in the background except MTMVT. Our tracker faithfully tracks the tiger, and obtains the best performance. Some examples are shown in Figure 5(D).

The *DH*, *Bolt*, *Skating1* and *Gym* sequences track fast moving human bodys in sports scenarios. In the *DH* sequence, L1T and IVT lose the target because of the distracting background and fast motion. MIL loses the target when the illumination changes suddenly. In the *Bolt*, *Skating1* and *Gym* sequences, the poses of targets changes rapidly and the appearance deforms frequently, which make them more challenging for existing trackers. Both L1T and IVT fail on all the sequences. MTT loses the targets soon on the *Bolt* and *Gym* sequences due to the deformation of the targets. VTD succeeds in *Bolt* and *Gym* because of the benefit of multiple types of features but drifts apart from the target at end of the *Skating1* sequence. However, only MTMVT successfully tracks all these targets in our experiments, which indicates the proposed tracker is not as sensitive to shape deformation as previous single view trackers, due to the effective use of the complementary features.

### 4.3. Quantitative Comparison

To quantitatively evaluate the performance of each tracker, we calculate the distance between the centers of the tracking result and the ground truth for each frame and plot these center location errors versus frame numbers, as done in [1] and [35]. Due to space limitation, we only show the error plots for eight sequences here and provide those for all twelve sequences in *supplementary material*. For a perfect result, the position error should be zero. As shown in Figure 6, the error plots of our tracker are generally lower than those of other trackers. This implies that our tracker outperforms other trackers on the test sequences. For a more intuitive comparison, the average position errors for the twelve sequences are summarized in Table 1. This shows that our tracker achieves the best average performance over all tested sequences.

Table 1. **Average position error** (pixels). Bold number in red indicates the best performance, while green indicates the second best. Note that the average is computed as a weighted average with respect to the sequence length.

| | L1T | MTT | VTD | MIL | IVT | MTMVT | MTT+O | MTMVT−O |
|---|---|---|---|---|---|---|---|---|
| Animal | 23.1 | 7.3 | 13.9 | **3.7** | 143.9 | **6.9** | 10.0 | 7.8 |
| Car4 | 5.7 | 2.0 | 28.8 | 59.4 | **1.4** | **1.6** | 3.5 | 2.8 |
| David | 26.1 | 38.7 | 27.0 | 16.0 | 20.1 | **3.5** | 7.6 | 5.3 |
| Kitesurf | 34.6 | 43.6 | 102.3 | **3.3** | 64.5 | **4.2** | 10.8 | 5.0 |
| Shaking | 59.9 | 11.9 | 8.4 | 8.2 | 112.2 | **4.5** | 6.6 | 5.6 |
| Faceocc2 | 9.1 | 6.9 | 23.0 | 13.9 | **4.7** | **5.6** | 5.8 | 5.7 |
| Sylv | 16.2 | 7.1 | 23.9 | 17.7 | 27.9 | **3.2** | 6.9 | 5.1 |
| Tiger1 | 20.7 | 30.9 | 28.9 | 26.3 | 122.3 | **8.1** | 15.3 | 10.4 |
| Bolt | 197.4 | 74.8 | 9.3 | 14.2 | 158.8 | **6.0** | 21.7 | 11.4 |
| DH | 18.5 | 4.3 | **3.7** | 4.9 | 62.0 | **4.1** | 5.9 | 4.3 |
| Skating1 | 33.9 | 6.6 | 34.2 | 41.4 | 53.9 | **4.7** | 5.2 | 25.3 |
| Gym | 93.8 | 71.8 | **5.9** | 25.4 | 32.8 | **7.3** | 11.9 | 6.9 |
| Average | 41.5 | 24.5 | 20.6 | 23.1 | 48.7 | **4.7** | 8.4 | 7.4 |

**Outlier Handling Performance:** To illustrate improvement of the proposed outlier handling method which including the introduction of auxiliary matrix $Q$ and the outlier rejection scheme presented in Section 3.3. We implement a Multi-Task Tracker with Outlier handling (MTT+O) using the robust multi-task sparse representation presented in Section 3.2, but let $K = 1$ using intensity feature only. We also implement a Multi-Task Multi-View Tracker without Outlier handling (MTMVT−O) using the representation presented in Section 3.2 but removing auxiliary matrix $Q$. As shown in Table 1, MTT+O demonstrates overall better performance comparing to the original MTT, while MTMVT shows its superiority comparing to MTMVT−O in terms of average error. Experimental results suggest that outliers should be specifically considered in mutli-task learning.

## 5. Conclusion

In this paper, we have presented a robust multi-task multi-view joint sparse learning method for particle filter-based tracking. By appropriately introducing the $l_{1,2}$ norm regularization, the method not only exploits the underlying relationship shared by different views and different particles, but also captures the frequently emerging outlier tasks which have been ignored by previous works. We implemented our method using four types of complementary features, i.e. intensity, color histogram, HOG and LBP, and extensively tested it on numerous challenging sequences. The experimental results demonstrate that the proposed method is capable of taking advantage of multi-view data and correctly handling the outlier tasks. Compared to five

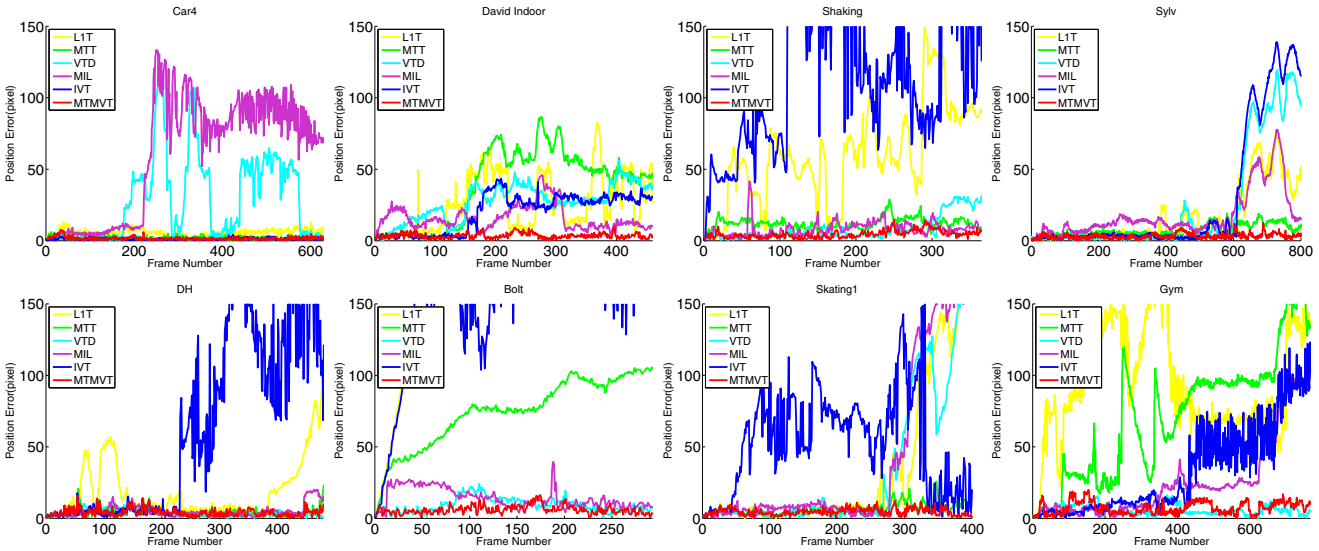


Figure 6. Location Error (in pixel) plot of each tracker on eight test sequences for quantitative comparison.

popular trackers, our tracker demonstrates superior performance. Moreover, the proposed method can potentially be extended to handle data obtained from sensors other than cameras.

## Acknowledgement

The first and the last co-authors of this paper were supported by the Australian Research Council under Discovery Project ARC DP-120103730.

## References

[1] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *TPAMI*, 33(8):1619–1632, 2011.

[2] V. Badrinarayanan, P. Perez, F. Le Clerc, and L. Oisel. Probabilistic color and adaptive multi-feature tracking with dynamically switched priority between cues. *ICCV*, 1–8, 2007.

[3] X. Chen, W. Pan, J. Kwok, and J. Carbonell. Accelerated gradient method for multi-task sparse learning problem. *ICDM*, 746–751, 2009.

[4] R. T. Collins, Y. Liu, and M. Leordeanu. Online selection of discriminative tracking features. *TPAMI*, 27(10):1631–1643, 2005.

[5] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *TPAMI*, 25(5):564–577, 2003.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *CVPR*, 1:886–893, 2005.

[7] W. Du and J. Piater. A probabilistic approach to integrating multiple cues in visual tracking. *ECCV*, 225–238, 2008.

[8] M. Godec, P. M. Roth, and H. Bischof. Hough-based tracking of non-rigid objects. *ICCV*, 81–88, 2011.

[9] P. Gong, J. Ye, and C. Zhang. Robust multi-task feature learning. *ICDM*, 895–903, 2012.

[10] Z. Hong, X. Mei, and D. Tao. Dual-force metric learning for robust distracter-resistant tracker. *ECCV*, 513–527, 2012.

[11] M. Isard and A. Blake. Condensation-conditional density propagation for visual tracking. *IJCV*, 29:5–28, 1998.

[12] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust online appearance models for visual tracking. *TPAMI*, 25(10):1296–1311, 2003.

[13] Z. Kalal, J. Matas, and K. Mikolajczyk. P-n learning: Bootstrapping binary classifiers by structural constraints. *CVPR*, 49–56, 2010.

[14] J. Kwon and K. M. Lee. Visual tracking decomposition. *CVPR*, 1269–1276, 2010.

[15] H. Li, C. Shen, and Q. Shi. Real-time visual tracking using compressive sensing. *CVPR*, 1305–1312, 2011.

[16] B. Liu, J. Huang, L. Yang, and C. Kulikowsk. Robust tracking using local sparse appearance model and k-selection. *CVPR*, 1313–1320, 2011.

[17] J. Liu, S. Ji, and J. Ye. Multi-task feature learning via efficient $\ell_{2,1}$-norm minimization. *Conf. on UAI*, 339–348, 2009.

[18] W. Liu and D. Tao. Multiview hessian regularization for image annotation. *TIP*, 22(7):2676–2687, 2013.

[19] X. Mei and H. Ling. Robust visual tracking and vehicle classification via sparse representation. *TPAMI*, 33(11):2259–2272, 2011.

[20] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai. Efficient minimum error bounded particle resampling l1 tracker with occlusion detection. *TIP*, 22(7):2661–2675, 2013.

[21] F. Moreno-Noguer, A. Sanfeliu, and D. Samaras. Dependent multiple cue integration for robust tracking. *TPAMI*, 30(4):670–685, 2008.

[22] Y. Nesterov. Gradient methods for minimizing composite objective function. *Center for Operations Research and Econometrics (CORE), Universit catholique de Louvain*, 76, 2007.

[23] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *TPAMI*, 24(7):971–987, 2002.

[24] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan. Locally orderless tracking. *CVPR*, 1940–1947, 2012.

[25] N. Quadrianto and C. H. Lampert. Learning multi-view neighborhood preserving projections. *ICML*, 425–432, 2011.

[26] D. Ross, J. Lim, R. Lin, and M. Yang. Incremental learning for robust visual tracking. *IJCV*, 77(1):125–141, 2008.

[27] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *TIP*, 19(6):1635–1650, 2010.

[28] O. Williams, A. Blake, and R. Cipolla. Sparse bayesian learning for efficient visual tracking. *TPAMI*, 27(8):1292–1304, 2005.

[29] T. Xia, D. Tao, T. Mei, and Y. Zhang. Multiview spectral embedding. *TSMC, Part B*, 40(6):1438–1446, 2010.

[30] C. Xu, D. Tao, and C. Xu. A survey on multi-view learning. *CoRR*, abs/1304.5634, 2013.

[31] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Comput.Surv.*, 38(4):13–45, 2006.

[32] J. Yoon, D. Kim, and K.-J. Yoon. Visual tracking via adaptive tracker selection with multiple features. *ECCV*, 28–41, 2012.

[33] X.-T. Yuan and S. Yan. Visual classification with multi-task joint sparse representation. *CVPR*, 3493–3500, 2010.

[34] K. Zhang, L. Zhang, and M.-H. Yang. Real-time compressive tracking. *ECCV*, 864–877, 2012.

[35] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via multi-task sparse learning. *CVPR*, 2042–2049, 2012.