Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Robust object tracking with RGBD-based sparse learning*#

Zi-ang MA[1], Zhi-yu XIANG‡[2]

([1]*College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China*)

([2]*Zhejiang Provincial Key Laboratory of Information Network Technology, Hangzhou 310027, China*)

E-mail: kobebean@zju.edu.cn; xiangzy@zju.edu.cn

Received June 14, 2016; Revision accepted Sept. 26, 2016; Crosschecked July 13, 2017

Abstract: Robust object tracking has been an important and challenging research area in the field of computer vision for decades. With the increasing popularity of affordable depth sensors, range data is widely used in visual tracking for its ability to provide robustness to varying illumination and occlusions. In this paper, a novel RGBD and sparse learning based tracker is proposed. The range data is integrated into the sparse learning framework in three respects. First, an extra depth view is added to the color image based visual features as an independent view for robust appearance modeling. Then, a special occlusion template set is designed to replenish the existing dictionary for handling various occlusion conditions. Finally, a depth-based occlusion detection method is proposed to efficiently determine an accurate time for the template update. Extensive experiments on both KITTI and Princeton data sets demonstrate that the proposed tracker outperforms the state-of-the-art tracking algorithms, including both sparse learning and RGBD based methods.

Key words: Object tracking; Sparse learning; Depth view; Occlusion templates; Occlusion detection
http://dx.doi.org/10.1631/FITEE.1601338 **CLC number:** TP391

1 Introduction

Visual object tracking is currently an important and fundamental research area in computer vision, with applications in many fields such as surveillance, human-computer interaction, and autonomous recognition systems (Yilmaz *et al.*, 2006; Wu *et al.*, 2013). Recently, the sparse representation based framework has emerged as an efficient solution for visual tracking (Candes *et al.*, 2006; Donoho, 2006; Mei and Ling, 2009; 2011; Ling *et al.*, 2010; Liu *et al.*, 2010). In Mei and Ling (2009), a tracking candidate was sparsely represented as a linear combination of the over-complete dictionary including target and trivial templates. Considering the interdependencies among particles, a multi-task sparse learning framework enforcing joint sparsity was proposed (Zhang TZ *et al.*, 2012b). In Hong *et al.* (2013) and Lan *et al.* (2014), multiple features, including color, edge, and texture, were used to complement the intensity for robust appearance modeling. Most of the existing sparse learning based trackers focus on the color image for its rich information. However, sparse tracking remains a challenging task, especially under conditions of varying illumination and extensive occlusions.

With the emergence of affordable depth sensors, such as lidar, radar, and stereo vision systems, range data has been widely applied in visual tracking (Choi *et al.*, 2011; Luber *et al.*, 2011; Song and Xiao, 2013). It has the potential of less sensitivity to varying illumination conditions and the capability of distinguishing the tracking target from the background. In Luber *et al.* (2011), multiple people were detected and tracked in RGBD data by combining online learning of the target appearance models with multiple hypothesis tracking. In Song and Xiao (2013),

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. 61571390) and the Fundamental Research Funds for the Central Universities, China (No. 2016QNA5004)
# A preliminary version was presented at the SAI Intelligent Systems Conference, Nov. 10–11, 2015, London, UK

ORCID: Zi-ang MA, http://orcid.org/0000-0001-8241-5303

RGBD-based features were used to build a more stable discriminative model and effectively classify potential target regions from the background. While the existing RGBD-based approaches perform well in tackling the issue of changing illumination, they often fail in complex environments.

One of the unresolved challenges for object tracking is the presence of occlusions. Various approaches have been proposed to address this problem, one of which is the trivial coefficient based method widely used in sparse tracking (Wright *et al.*, 2009; Yang and Zhang, 2010; Mei *et al.*, 2011). In the sparse representation stage, trivial templates are activated when a candidate particle cannot be represented well by the target template set. Thus, the nonzero entries within the trivial coefficients indicate the corresponding areas occluded in particle observation. Meanwhile, the RGBD-based trackers identify occlusions when, in the depth histogram of the target region, a newly rising peak emerges with a smaller value than that of the target (Song and Xiao, 2013). The existing methods for occlusion detection have proven effective. However, their accuracy usually decreases considerably with extensive and successive occlusions.

Inspired by the advances discussed above, to improve the tracking performance under conditions of varying illumination and heavy occlusions, we propose an RGBD-based sparse learning tracker. Three respects of the existing sparse tracking framework are greatly strengthened:

1. Depth view. For robust appearance modeling, the depth feature is integrated to replenish previous color image based visual features as an independent kind of view.

2. Occlusion template set. The segmented occlusion area from the target region is formulated as an occlusion template set. Integrating these occlusion templates with the existing over-complete dictionary endows the tracker with the ability to handle various occlusion conditions.

3. Occlusion detection. A depth-based histogram analysis is employed for effectively detecting occlusions and determining the proper time of template update. This study is an extension of our previous work (Ma and Xiang, 2015). To the best of our knowledge, it is the first work that designs special templates for handling various extreme occlusion conditions. It covers the gap between sparse learning based tracking and traditional RGBD-based tracking, and it circumvents the drawbacks of both approaches. Fig. 1 illustrates the framework of the proposed tracker.

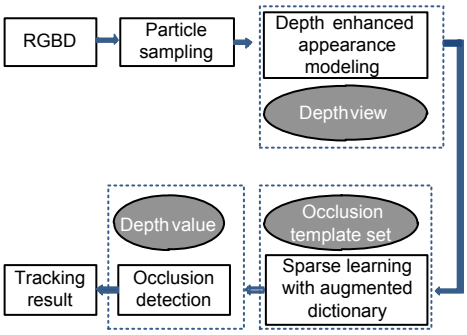

Fig. 1 Framework of the proposed tracker

The dashed bounding boxes illustrate how the valuable information provided by the range data (denoted as ellipses) is enforced in the tracker

2 Related work

Various remarkable methods for visual tracking have been presented over the years, and can be divided into two categories: discriminative and generative. In discriminative methods, a tracking problem is regarded as a binary classification problem to locate the target region that best separates the target from the background. Examples of discriminative tracking methods are ensemble tracking (Avidan, 2007), online multiple instance learning (Babenko *et al.*, 2009), sparse Bayesian learning (Williams *et al.*, 2005), and global mode seeking (Yin and Collins, 2008). Generative methods formulate the tracking problem as a search for a potential target location that is most similar in appearance to the generative model. Examples include eigen tracking (Black and Jepson, 1998), mean shift tracking (Comaniciu *et al.*, 2003), covariance tracking (Porikli *et al.*, 2006), and incremental tracking (Ross *et al.*, 2008).

With the increasing popularity of affordable depth sensors, the range data is widely used in visual tracking for its ability to provide robustness to occlusions and to be less sensitive to varying illumination (Choi *et al.*, 2011; Luber *et al.*, 2011; Song and Xiao, 2013). In Choi *et al.* (2011) and Luber *et al.* (2011), people can be detected and tracked from the

image and depth sensors via multi-detector fusion. The RGBD HOG feature was used to build a more stable discriminative model and effectively classify potential target regions (Song and Xiao, 2013). RGBD+OF, which combines RGBD HOG feature detection with optical flow based tracking (Zhang, 1994), and RGBDOcc+OF, which further considers occlusion handling, were proven to be the best trackers using RGBD data based on tracking by detection.

On the other hand, sparse representation has been widely exploited in visual object tracking, as well as for recognition and classification. In Mei and Ling (2009), each target candidate was represented as a sparse linear combination of the template set by solving an l_1-regularized least squares problem. To improve the real-time performance, an efficient L1 tracker (L1T) with a two-stage bounded resampling scheme as well as occlusion detection was introduced in Mei *et al.* (2011). In Bao *et al.* (2012), an accelerated proximal gradient approach was developed to guarantee quadratic convergence to hasten L1T without loss of precision. In addition, visual tracking was formulated as a multi-task sparse learning based tracker (MTT) considering the underlying interdependencies between particles (Zhang TZ *et al.*, 2012b). Employing multiple visual features including color, edge, and texture to replenish intensity in appearance modeling, the multi-task multi-view tracker (MTMVT) achieved a more robust performance (Hong *et al.*, 2013).

Sparse learning based trackers can be formulated as a minimization problem of the reconstruction error constrained by regularizations. As a common regularization, the row sparse constraint forces all particles into being jointly represented by as few templates as possible in MTT (Zhang TZ *et al.*, 2012b). The subsequent sparse learning based trackers strengthen MTT in two regards:

1. Added regularizations. The LRST proposed in Zhang TZ *et al.* (2012a) extends MTT by exploiting the underlying low-rank constraint. In Zhang *et al.* (2015a), LRST was further extended by formulating object tracking as a low-rank, sparse, and temporally consistent representation problem. The column sparse constraint imposed in MTMVT (Hong *et al.*, 2013) enables the capture of outlier particles, which are sampled far away from the tracking target.

2. Structure information. Considering the pairwise structural correlations between particles, the S-MTT presented in Zhang *et al.* (2013) expands MTT by imposing the structure information via graph regularization. To employ the spatial layout structure of the target region, the STT proposed in Zhang *et al.* (2015b) samples several local image patches inside each candidate region and target template with the same spatial layout. Thus, the row sparse constraint forces all particles into sharing the same dictionary basis over all local image patches.

3 RGBD and the sparse learning based tracker

3.1 Particle filter

The particle filter indicates the main regions distributed around a tracking target with multiple discrete particles, and it is used primarily for robust tracking under complex environments. In a sparse tracking framework, $\boldsymbol{x}_t$ denotes the state variable of the sampled particle, describing the target's affine motion at time t with four deformation and two translation parameters (Pei and Lin, 1995). Given all available observations $\boldsymbol{z}_{1:t}=\{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_t\}$, the tracking problem can be formulated as an estimation of the posterior distribution of $\boldsymbol{x}_t$, which is recursively updated using the following expression:

$$p(\boldsymbol{x}_t \mid \boldsymbol{z}_{1:t}) = \frac{p(\boldsymbol{z}_t \mid \boldsymbol{x}_t)p(\boldsymbol{x}_t \mid \boldsymbol{z}_{1:t-1})}{p(\boldsymbol{z}_t \mid \boldsymbol{z}_{1:t-1})} = \frac{p(\boldsymbol{z}_t \mid \boldsymbol{x}_t)\int p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})p(\boldsymbol{x}_{t-1} \mid \boldsymbol{z}_{1:t-1})\mathrm{d}\boldsymbol{x}_{t-1}}{p(\boldsymbol{z}_t \mid \boldsymbol{z}_{1:t-1})}, \tag{1}$$

where the state transition distribution $p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})$ is modeled to be Gaussian, and $p(\boldsymbol{z}_t \mid \boldsymbol{x}_t)$ indicates the observation likelihood. By approximating the posterior $p(\boldsymbol{x}_{t-1} \mid \boldsymbol{z}_{1:t-1})$ with a set of particles $\{\boldsymbol{x}_{t-1}^i, w_{t-1}^i\}_{i=1}^N$, Eq. (1) can be rewritten as

$$p(\boldsymbol{x}_t \mid \boldsymbol{z}_{1:t}) = \frac{p(\boldsymbol{z}_t \mid \boldsymbol{x}_t)\sum_{i=1}^N w_{t-1}^i p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}^i)}{p(\boldsymbol{z}_t \mid \boldsymbol{z}_{1:t-1})}, \tag{2}$$

where w_{t-1}^i indicates the importance weight of particle $\boldsymbol{x}_{t-1}^i$. Particles are updated and resampled in each

frame, and in the case of the bootstrap filter, the updated weight w_{t-1}^{i} is proportional to the corresponding observation likelihood $p(\boldsymbol{z}_{t-1} \mid \boldsymbol{x}_{t-1}^{i})$.

To model the observation likelihood $p(\boldsymbol{z}_t \mid \boldsymbol{x}_t)$, a region of interest is first cropped from the image using $\boldsymbol{x}_t$ as the parameter and normalized to the template size. Multiple features are then extracted from the region and treated as a candidate particle observation. The observation likelihood $p(\boldsymbol{z}_t \mid \boldsymbol{x}_t)$ indicates the similarity between a candidate particle observation and the dictionary templates. In this study, $p(\boldsymbol{z}_t \mid \boldsymbol{x}_t)$ is formulated using a reconstruction error in the sparse representation stage described below.

3.2 Depth-enhanced appearance modeling

An ideal appearance model should be adaptive to intrinsic variations such as varying posture, as well as to extrinsic variations such as changing illumination and various occlusion conditions (Yilmaz *et al.*, 2006). Existing color image based features have been shown to be liable to failure with varying and complex illumination. Considering the emergence of affordable depth sensors, the depth images of a scene are easily obtained. The depth image is insensitive to illumination change and also provides valuable 3D information for tracking. Therefore, the color image based features are enhanced with a depth view to provide a more comprehensive appearance modeling in the proposed tracker.

3.3 Occlusion template set

For each view index m=1, 2, …, M, the target template set (denoted as $\boldsymbol{T}_t^m \in \mathbb{R}^{d_m \times n_t}$, $d_m \gg n_t$) is initialized by slightly shifting the selected target region in the first frame to all directions, and is updated when necessary in the following sequences. Each column in $\boldsymbol{T}_t^m$ represents a target template obtained by extracting the mth feature as a column vector, and d_m and n_t represent the dimension of the mth feature and the number of target templates, respectively. The target templates $\boldsymbol{T}_t^m$ are concatenated with the trivial templates $\boldsymbol{I}_{d_m}$ ($d_m \times d_m$ identity matrix) to construct the over-complete dictionary.

Nevertheless, while the online updating strategy endows the existing dictionary with the ability to evolve quickly to accommodate a new environment, it is still insufficient for handling heavy occlusions. Normally, when an obvious occlusion is detected, it will not go away for a certain period of time. With a depth-based occlusion detection method (see Section 4), the occlusion area segmented from the previous tracking result is formulated as a model to generate the occlusion template set, which can effectively handle various extreme occlusion conditions in the next frame.

If an apparent occlusion is detected in the previous tracking result, it will be segmented and then regularized into a rectangular region. This normalized region is then depicted with the highest proportion of color extracted from the corresponding color image area as an occlusion model for color. Likewise, the average depth value calculated from the corresponding depth image area is depicted as the model for depth. These occlusion models are gradually expanded in both width and height dimensions until they are twice the initial size and are compressed until half the size. The nearest tracking result without occlusion is preserved as the background of the generated occlusion templates. By traversing the occlusion models with various sizes to all directions on the background, the occlusion template set for the current frame can be obtained as $\boldsymbol{O}_t^m \in \mathbb{R}^{d_m \times n_o}$. This generative process of n_o occlusion templates is illustrated in Fig. 2.

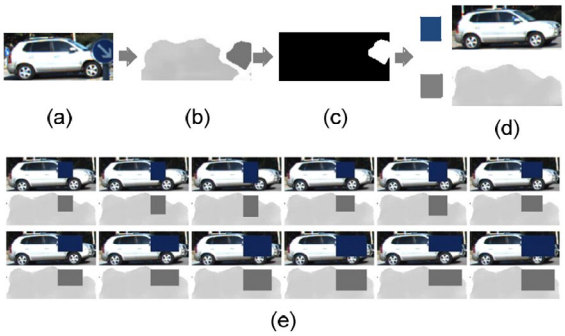

Fig. 2 An example from the training_01 sequence in the KITTI Vision benchmark to illustrate the generation of occlusion templates: (a) tracking result with an apparent occlusion; (b) corresponding depth image; (c) binarization of the target region, where occlusions are expressed as white pixels and the remaining regions are black; (d) occlusion models for color and depth images (left) and their corresponding traversed backgrounds (right); (e) occlusion templates generated by traversing occlusion models with various sizes to all directions on the corresponding backgrounds, in which only representative ones are presented (References to color refer to the online version of this figure)

The existing over-complete dictionary will be augmented with the occlusion templates when the previous tracking result is detected as largely occluded. Thus, the occlusion templates exist only in partial frames rather than the whole tracking sequence. In contrast to the update strategy for target templates, the occlusion template set is completely rebuilt as necessary. The effectiveness of the augmented dictionary for tracking is illustrated in Fig. 3.

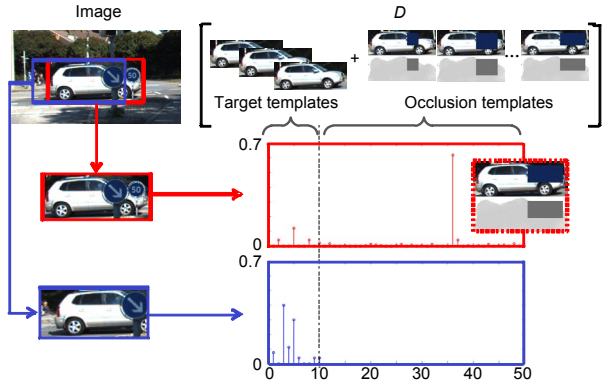

Fig. 3 The effectiveness of the generated occlusion template set for tracking (References to color refer to the online version of this figure)

The red and blue bounding boxes represent tracking results with and without occlusion templates integrated into the augmented dictionary, respectively. The blue one tends to contain partial target area and much more background in the bounding box. The dashed bounding box represents the zoomed-in snapshot of the occlusion template corresponding to the highest sparse coefficient value. It is proven that the proposed occlusion templates facilitate accurate tracking even under heavy occlusions

3.4 Sparse learning with the augmented dictionary

The generated occlusion templates $\boldsymbol{O}_t^m$ are combined with the existing over-complete dictionary to construct an augmented dictionary $\boldsymbol{D}_t^m=[\boldsymbol{T}_t^m, \boldsymbol{O}_t^m, \boldsymbol{I}_{d_m}]$. Given that each particle shares dependencies with others, N candidate particle observations are considered to be a joint linear combination of the augmented dictionary, such that

$$\boldsymbol{X}_t^m = \boldsymbol{D}_t^m \cdot \boldsymbol{C}_t^m = \left[\boldsymbol{T}_t^m, \boldsymbol{O}_t^m, \boldsymbol{I}_{d_m}\right] \cdot \begin{bmatrix} \boldsymbol{C}_{\mathrm{T}}^m \\ \boldsymbol{C}_{\mathrm{O}}^m \\ \boldsymbol{C}_{\mathrm{I}}^m \end{bmatrix} \in \mathbb{R}^{d_m \times N}, \quad (3)$$

where $\boldsymbol{C}_t^m \in \mathbb{R}^{(n_{\mathrm{t}}+n_{\mathrm{o}}+d_m)\times N}$ represents the coefficient matrix of sparse representation for the mth view. $\boldsymbol{C}_t^m$ is composed of target coefficients $\boldsymbol{C}_{\mathrm{T}}^m$, occlusion coefficients $\boldsymbol{C}_{\mathrm{O}}^m$, and trivial coefficients $\boldsymbol{C}_{\mathrm{I}}^m$.

Considering the underlying relationship between particles, the corresponding particle observations for different views should share the same dictionary basis. Therefore, a group lasso penalty l_{12} is imposed on the global coefficient matrix $\boldsymbol{C}_t=[\boldsymbol{C}_t^1, \boldsymbol{C}_t^2, \ldots, \boldsymbol{C}_t^M]$ to capture the shared templates among all tasks over all views. In other words, $\boldsymbol{C}_t$ is regularized by the row sparse constraint, which assumes that all particles jointly share as few templates as possible. Thus, $\boldsymbol{C}_t$ is formulated as a minimum reconstruction error through a regularized l_{12} minimization function as

$$\min_{\boldsymbol{C}} \sum_{m=1}^{M} \left\| \boldsymbol{D}_t^m \boldsymbol{C}_t^m - \boldsymbol{X}_t^m \right\|_{\mathrm{F}}^2 + \lambda \left\| \boldsymbol{C}_t \right\|_{1,2} \quad (4)$$

with

$$\left\| \boldsymbol{C} \right\|_{p,q} = \left(\sum\nolimits_i \left(\sum\nolimits_j |c_{ij}|^q \right)^{\frac{p}{q}} \right)^{\frac{1}{p}}, \quad (5)$$

where λ controls the tradeoff between the reconstruction error and the row sparse constraint. Then, the observation likelihood for each $\boldsymbol{x}_t^i$ is given as

$$p(\boldsymbol{z}_t \mid \boldsymbol{x}_t^i) = \frac{1}{\Gamma} \exp\left\{ -\alpha \sum_{m=1}^{M} \left\| \boldsymbol{T}_t^m \boldsymbol{C}_i^{\mathrm{T}} + \boldsymbol{O}_t^m \boldsymbol{C}_i^{\mathrm{O}} - \boldsymbol{X}_i^m \right\|_2^2 \right\}, \quad (6)$$

where α is a constant controlling the shape of the Gaussian kernel, and Γ is a normalization factor. $\boldsymbol{C}_i^m$, $\boldsymbol{C}_i^{\mathrm{O}}$, and $\boldsymbol{X}_i^m$ represent the ith column of matrices $\boldsymbol{C}_{\mathrm{T}}^m$, $\boldsymbol{C}_{\mathrm{O}}^m$, and $\boldsymbol{X}_t^m$, respectively.

Finally, the optimal particle $\boldsymbol{x}_t^*$ with the maximum observation likelihood is chosen as the current tracking result:

$$\boldsymbol{x}_t^* = \arg\max_{\boldsymbol{x}_t^i \in S_t} p(\boldsymbol{z}_t \mid \boldsymbol{x}_t^i), \quad (7)$$

where S_t is the set of particles.

3.5 Optimization algorithm

Note that objective function (3) is composed of a differentiable convex function and a non-smooth but

convex regularization. For efficient tracking, the accelerated proximal gradient (APG) method (Chen *et al.*, 2009; Bao *et al.*, 2012) is exploited to solve the regularized minimization problem with a quadratic convergence rate. Each APG iteration consists of two steps: (1) gradient mapping which updates the current representation matrix $\boldsymbol{C}^{(k)}$ with the aggregation matrix $\boldsymbol{Z}^{(k)}$ fixed; (2) an aggregation step to update $\boldsymbol{Z}^{(k)}$.

Step 1: Update $\boldsymbol{C}^{(k+1)}$. Update of the representation matrix $\boldsymbol{C}^{(k+1)}$ is formulated as an l_{12}-norm minimization problem:

$$\boldsymbol{C}^{(k+1)} = \arg\min_{\boldsymbol{Y}} \frac{1}{2}\|\boldsymbol{Y}-\boldsymbol{H}\|_{\mathrm{F}}^2 + \eta\lambda\|\boldsymbol{Y}\|_{1,2}, \quad (8)$$

where

$$\boldsymbol{H} = \boldsymbol{Z}^{(k)} - 2\eta\sum_{m=1}^{M}(\boldsymbol{D}_t^m)^{\mathrm{T}}(\boldsymbol{D}_t^m\boldsymbol{Z}^{(k)} - \boldsymbol{X}_t^m),$$

and η is a parameter controlling the step penalty. An efficient closed-form solution can be attained via Eq. (8):

$$\boldsymbol{C}^{(k+1)} = G_{\eta\lambda}(\boldsymbol{H}), \quad (9)$$

where

$$G_{\lambda}(\boldsymbol{h}_i) = \max\left(0, 1-\lambda/\|\boldsymbol{h}_i\|_2\right)\boldsymbol{h}_i,$$

and $\boldsymbol{h}_i$ denotes the ith row of matrix $\boldsymbol{H}$.

Step 2: Update $\boldsymbol{Z}^{(k+1)}$. Update of the aggregation matrix $\boldsymbol{Z}^{(k+1)}$ is formulated as

$$\boldsymbol{Z}^{(k+1)} = \boldsymbol{C}^{(k+1)} + \alpha_{k+1}\frac{1-\alpha_k}{\alpha_k}\left(\boldsymbol{C}^{(k+1)} - \boldsymbol{C}^{(k)}\right), \quad (10)$$

where α_k=2/(k+3) for $k\geq 1$ and α_0=1.

Note that $\boldsymbol{C}^{(0)}$ and $\boldsymbol{Z}^{(0)}$ are set to zero matrices for initialization. The APG algorithm stops until the descent of the objective function is below a pre-defined threshold. For clarity, the proposed tracker is summarized in Algorithm 1.

Algorithm 1 RGBD-based sparse learning

Input: previous tracking result $\boldsymbol{F}_{t-1}$, previous particle set $S_{t-1}=\{\boldsymbol{x}_{t-1}^i\}_{i=1}^{N_0}$, and augmented dictionary $\boldsymbol{D}_t^m=[\boldsymbol{T}_t^m, \boldsymbol{O}_t^m, \boldsymbol{I}_{d_m}]$

Output: current tracking result $\boldsymbol{F}_t$, current particle set $S_t=\{\boldsymbol{x}_t^i\}_{i=1}^{N_0}$, and updated augmented dictionary $\boldsymbol{D}_{t+1}^m=[\boldsymbol{T}_{t+1}^m, \boldsymbol{O}_{t+1}^m, \boldsymbol{I}_{d_m}]$

1. Draw particles $\boldsymbol{x}_t^i$ from $\boldsymbol{x}_{t-1}^i$ (i=1, 2, …, N) with a Gaussian distribution
2. Crop candidate particle observations $\boldsymbol{X}_t^m$ from the image by applying an affine transformation using $\boldsymbol{x}_t^i$ as the parameter
3. Solve minimization problem (3) with the row sparse constraint
4. Calculate the observation likelihood for each particle according to Eq. (5) and weight each particle in S_t with the corresponding observation likelihood
5. Select the particle with the maximum observation likelihood as the current tracking result $\boldsymbol{F}_t$ according to Eq. (6)
6. Detect whether there is an apparent occlusion in the tracking result with the proposed method
7. Update the target template set from $\boldsymbol{T}_t^m$ to $\boldsymbol{T}_{t+1}^m$
8. Rebuild the occlusion template set $\boldsymbol{O}_{t+1}^m$ if an apparent occlusion is detected in step 6
9. Repeat steps 1–8 until the sequence ends

4 Depth-based occlusion detection

A fixed template set is insufficient for handling the appearance variations of the tracking target during the whole sequence. Thus, an online update strategy replaces the least important template with the current tracking result if none of the templates are similar to the target region (Mei and Ling, 2009). To avoid improper template changes, the tracking result with an apparent occlusion should not be added to the template set (Mei *et al.*, 2011). With the range data at hand, an efficient depth-based algorithm for occlusion detection is proposed to determine the accurate time of template update.

First, a depth-based histogram analysis for each tracking result is carried out. In particular, the depth values inside the bounding box are divided into p equally spaced bins as $\boldsymbol{b}^i=\{n_{\mathrm{b}}^i, x^i\}$ (i=1, 2, …, p), in which n_{b}^i and x^i denote the number of elements and the center position of the ith bin, respectively. The width of each bin is specified as w_{b}. Then, p equally spaced bins are clustered into q categories as $\boldsymbol{c}^j=\{\boldsymbol{b}^{j_1}, \boldsymbol{b}^{j_1+1}, \cdots, \boldsymbol{b}^{j_2}\}$, where j_1 and j_2 represent the indices of the initial and last bins in the jth category, respectively. Thus, the category with the largest number of elements, denoted as the kth category in Eq. (8), is

selected as the depth distribution of the tracking target. At frame t, the average depth of tracking target avg_t and the lower boundary of the depth distribution area lb_t can be obtained by

$$\text{avg}_t = \sum_{l=k_1}^{k_2} n_{\text{b}}^l \cdot x^l \Big/ \sum_{l=k_1}^{k_2} n_{\text{b}}^l, \tag{11}$$

$$\text{lb}_t = x^{k_1} - w_{\text{b}}. \tag{12}$$

Considering that the relative depth distribution of the tracking target does not vary too much between successive frames, the category with the largest number of elements near avg_t is selected as the depth distribution of the tracking target in the next frame.

Given the target's depth distribution at each frame, the pixels with an absolute depth value smaller than lb_t are identified as occlusions within the corresponding tracking result. By applying dilation and erosion operations to the binarization target region, it can be concluded that the tracking result is obscured if the largest connected white area is larger than a predefined proportion. As illustrated in Fig. 4, the proposed approach demonstrates a much better accuracy compared to the existing trivial coefficient based method.

Fig. 4 Performance comparison for occlusion detection: (a) tracking result with an apparent occlusion; (b) occlusion detection result using our proposed method; (c) occlusion detection result obtained by the trivial coefficient based method

5 Experimental results

In this section, the proposed tracker was compared with two popular trackers, i.e., compressive tracking (CT) (Zhang K *et al.*, 2012) and tracking with multiple instance learning (MIL) (Babenko *et al.*, 2009), and with the state-of-the-art sparse tracking algorithms, i.e., L1T (Mei and Ling, 2009), MTT (Zhang TZ *et al.*, 2012b), and MTMVT (Hong *et al.*, 2013), on the KITTI Vision benchmark (Fig. 5). Furthermore, since the range data was used, the tracker was compared with other RGBD-based tracking algorithms, i.e., RGBD+OF and RGBDOcc+OF (Song and Xiao, 2013), on the Princeton Tracking benchmark (Fig. 6). The challenging sequences from these two data sets contained complex scenes, including moving cameras, varying illumination and occlusion conditions, and variations in posture and scale. The experiments were conducted by running source codes provided by the authors.

5.1 Implementation details

All experiments were carried out on a PC with an Intel i7-4770 CPU (3.40 GHz) in MATLAB 2014a. L1T and MTT exploited the intensity for appearance modeling, while MTMVT employed other visual features such as color, histogram of oriented gradients (HOG) (Dalal and Triggs, 2005), and local binary patterns (LBP) (Ojala *et al.*, 2002) to complement the intensity in the appearance expression. In the proposed tracker, color image based features were enhanced with the depth view to provide a more comprehensive appearance model.

The parameters were selected as follows. The number of sampled particles was set as N=400. The number of target templates was set as n_{t}=10. The state transitional probability $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ was modeled by a zero-mean Gaussian with a diagonal covariance matrix with values of (0.03, 0.0005, 0.0005, 0.03, 4, 4). The tradeoff parameter was set to λ=0.5. The predefined proportion for occlusion detection was set as 20%.

Two evaluation criteria were applied to evaluate the overall tracking performance quantitatively, i.e., center position error (CPE) and success rate (SR) (Song and Xiao, 2013). CPE represents the Euclidean distance between the center of the tracking result and the ground truth. Each tracking result is concluded as correct when the overlapping ratio (OR) is larger than 0.5, where the measurement of OR is defined as

$$\text{OR} = \frac{\text{area}(B_{\text{T}} \cap B_{\text{G}})}{\text{area}(B_{\text{T}} \cup B_{\text{G}})}, \tag{13}$$

and B_{T} and B_{G} denote the bounding boxes of the tracking result and ground truth, respectively. Thus, SR represents the proportion of correct frames in the whole sequence.

Fig. 5 Comparisons of the tracking results on the KITTI Vision benchmark: (a) training_00 (frames 2, 31, 43, 47, 51, 59); (b) training_01 (frames 371, 375, 379, 383, 387, 392); (c) training_13 (frames 139, 150, 158, 162, 166, 171); (d) training_15 (frames 87, 96, 101, 103, 106, 108) (References to color refer to the online version of this figure)

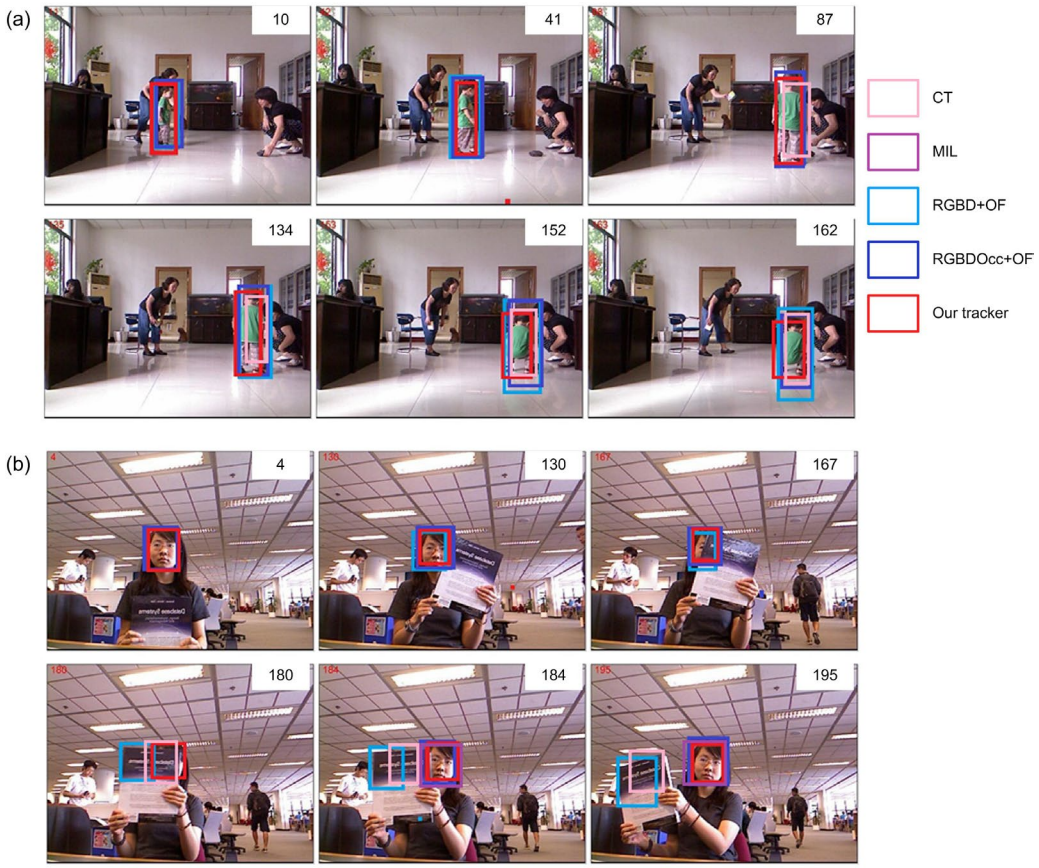

Fig. 6 Comparisons of the tracking results on the Princeton Tracking benchmark: (a) child_no1 (frames 10, 41, 87, 134, 152, 162); (b) face_occ5 (frames 4, 130, 167, 180, 184, 195) (References to color refer to the online version of this figure)

5.2 Comparisons on the KITTI Vision benchmark

The training_00 sequence shown in Fig. 5a tracked a moving van continuously undergoing posture variations, varying illuminations, and extensive occlusions. Only our proposed tracker succeeded in tracking over the whole sequence. According to Fig. 5a, L1T completely lost the target when the van's pose changed drastically, while MTT and MTMVT tended to drift away when the van ran into heavy occlusions. For the training_01 sequence (Fig. 5b), L1T, MTT, and MTMVT all failed in tracking when the target underwent quick illumination changes or consecutive occlusions. The depth-enhanced appearance modeling endowed our tracker with robustness when faced with varying illuminations. With the augmented dictionary, our tracker performed better when dealing with extensive and successive occlusions, as illustrated in Fig. 3.

A fast moving cyclist was tracked in sequences training_13 and training_15, as shown in Figs. 5c and 5d, respectively. In the training_13 sequence, L1T and MTT lost the target during the sudden illumination change, while MTMVT gradually drifted away when undergoing a drastic posture change. In the training_15 sequence, MTT and MTMVT tended to drift away in the presence of obstructions. The tracking performances on these two sequences show that our tracker demonstrates excellent effectiveness and robustness in both vehicle and pedestrian tracking.

To further highlight the benefit of our design in the occlusion template set, a preliminary RGBD-based MTMVT version (denoted as MTMVT+Depth), which enhanced MTMVT with a depth view, was compared with the proposed tracker. The experimental results show that MTMVT+Depth was indeed less sensitive to illumination compared with its predecessor. However, it is prone to drift under occlusion conditions (Figs. 5a, 5b, and 5d). Thus, comparisons with MTMVT+Depth further demonstrated the effectiveness of the proposed occlusion templates.

Quantitative evaluations on the KITTI Vision benchmark are illustrated in Table 1, where the best performance is marked in bold. The CPEs for each frame versus the frame indices are plotted in Figs. 7a–7d to provide a visual comparison.

5.3 Comparisons on the Princeton Tracking benchmark

Due to the use of range data, our tracker was also compared with some recent and most advanced RGBD-based methods to further verify its effectiveness. In the child_no1 sequence (Fig. 6a), all methods succeeded in tracking the boy in spite of various combined variations in pose and scale, while RGBDOcc+OF and RGBD+OF tended to include much redundant background area in the tracking results. For the face_occ5 sequence (Fig. 6b), CT and RGBD+OF gradually drifted away, while RGBDOcc+OF failed to predict the position of the tracking target when the girl's face was entirely occluded.

Quantitative evaluations on the Princeton Tracking benchmark are illustrated in Table 2 and Figs. 7e and 7f. Both qualitative and quantitative experimental results showed that our tracker obtained the best average performance compared to previous sparse learning and RGBD based tracking algorithms.

Table 1 Quantitative comparisons on the KITTI Vision benchmark in terms of the center position error and success rate

Sequence	Center position error (pixel)						Success rate					
	CT	MIL	L1T	MTT	MTMVT	Our tracker	CT	MIL	L1T	MTT	MTMVT	Our tracker
training_00	62.17	84.62	79.42	38.56	35.96	**7.86**	0.69	0.58	0.68	0.70	0.72	**0.94**
training_01	227.41	186.12	331.11	76.74	84.46	**6.18**	0.26	0.32	0.17	0.74	0.62	**0.96**
training_13	99.84	98.72	152.08	97.70	49.33	**12.26**	0.49	0.52	0.26	0.32	0.54	**0.92**
training_15	24.08	32.29	117.59	33.62	31.26	**11.80**	**1.00**	0.76	0.41	0.73	0.77	**1.00**
training_18	13.81	9.82	4.68	5.10	5.24	**3.96**	**1.00**	0.98	**1.00**	**1.00**	**1.00**	**1.00**
training_20	29.53	17.64	3.12	3.43	**2.16**	2.76	0.76	0.86	0.98	0.98	**0.99**	0.98
Average	78.31	71.54	114.67	42.53	34.70	**7.48**	0.70	0.68	0.58	0.75	0.77	**0.96**

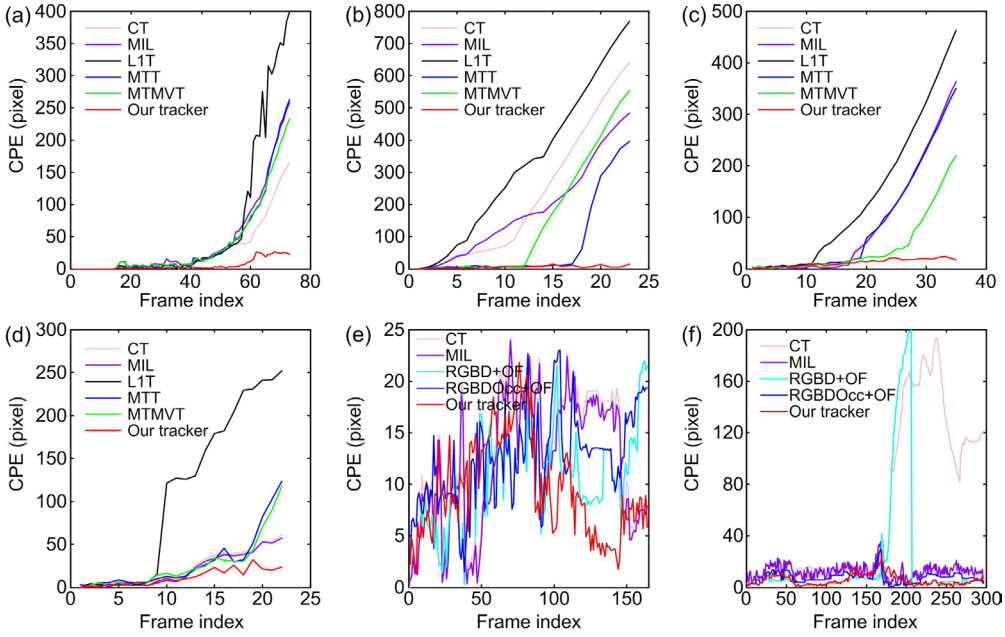

Fig. 7 Quantitative frame-by-frame comparisons in terms of the center position error (CPE): (a)–(d) are the CPEs of the training_00, training_01, training_13, and training_15 sequences in the KITTI benchmark, respectively; (e) and (f) are the CPEs of the child_no1 and face_occ5 sequences in the Princeton benchmark, respectively (References to color refer to the online version of this figure)

Table 2 Quantitative comparisons on the Princeton Tracking benchmark in terms of the center position error and success rate

Sequence	Center position error (pixel)					Success rate				
	CT	MIL	RGBD+ OF	RGBDOcc+ OF	Our tracker	CT	MIL	RGBD+ OF	RGBDOcc+ OF	Our tracker
child_no1	13.16	12.54	10.76	12.51	**9.56**	1.00	0.98	0.96	1.00	**1.00**
zcup_move1	15.16	20.74	7.22	7.74	**4.38**	0.89	0.92	1.00	0.96	**1.00**
face_occ5	69.73	9.68	19.43	7.75	**5.86**	0.51	0.88	0.85	0.92	**0.96**
newex_occ4	103.37	72.91	83.63	56.10	**20.68**	0.56	0.58	0.50	0.68	**0.88**
Average	50.34	28.97	30.26	34.70	**10.12**	0.74	0.84	0.83	0.89	**0.96**

5.4 Discussion

One of the problems for sparse trackers is time consumption, which entails solving the minimization problem with the lasso penalty. The average running time of MTMVT using MATLAB is about 3.76 s/frame with 400 particles and 10 target templates. For the proposed tracker, the particle pretreatment strategy presented in our previous work (Ma and Xiang, 2015) was carried out to prune stray particles before solving the computationally expensive objective function. Thus, the average running time was reduced to 0.96 s/frame with the same implementation in our proposed tracker. Considering that the computational cost grows linearly with the number of sampled particles, the tradeoff quantified by N between the tracking performance and efficiency is summarized in Table 3. Accordingly, an empirical value of N=400 was selected for particle sampling.

While our proposed tracker performs exceedingly well in tackling conditions of varying illuminations and occlusions, it often fails in complex environments such as drastic posture change. Therefore, future work will focus on this limitation and on how to further reduce the time consumption.

Table 3 Tradeoff quantified by N between tracking performance and efficiency on the KITTI Vision benchmark

Number of sampled particles, N	Average time consumption (s)	Average CPE (pixel)
100	0.24	58.81
200	0.47	40.57
300	0.74	37.14
400	0.96	8.54
500	1.21	8.26
600	1.44	7.96

CPE: center position error

6 Conclusions

In this paper, a robust and efficient RGBD and sparse learning based tracking algorithm is proposed. The new algorithm greatly improves the tracking performance by efficiently exploiting the range data from affordable depth sensors. First, the existing color image based features are enhanced with an extra depth view to provide a more comprehensive appearance model. By augmenting the existing dictionary with an occlusion template set, the proposed tracker facilitates accurate tracking under various extreme occlusion conditions. Finally, a depth-based algorithm for detecting occlusions is further proposed to efficiently determine the proper time for the template update. Qualitative and quantitative evaluations of various challenging sequences show that the proposed tracker outperforms the state-of-the-art tracking algorithms, including the sparse learning and RGBD based methods.

References

Avidan, S., 2007. Ensemble tracking. *IEEE Trans. Patt. Anal. Mach. Intell.*, **29**(2):261-271. https://doi.org/10.1109/TPAMI.2007.35

Babenko, B., Yang, M.H., Belongie, S., 2009. Visual tracking with online multiple instance learning. IEEE Conf. on Computer Vision and Pattern Recognition, p.983-990. https://doi.org/10.1109/CVPR.2009.5206737

Bao, C.L., Wu, Y., Ling, H.B., *et al.*, 2012. Real time robust L1 tracker using accelerated proximal gradient approach. IEEE Conf. on Computer Vision and Pattern Recognition, p.1830-1837. https://doi.org/10.1109/CVPR.2012.6247881

Black, M.J., Jepson, A.D., 1998. EigenTracking: robust matching and tracking of articulated objects using a view-based representation. *Int. J. Comput. Vis.*, **26**(1): 63-84. https://doi.org/10.1023/A:1007939232436

Candes, E.J., Romberg, J.K., Tao, T., 2006. Stable signal recovery from incomplete and inaccurate measurements. *Commun. Pure Appl. Math.*, **59**(8):1207-1223. https://doi.org/10.1002/cpa.20124

Chen, X., Pan, W.K., Kwok, J.T., *et al.*, 2009. Accelerated gradient method for multi-task sparse learning problem. 9th IEEE Int. Conf. on Data Mining, p.746-751. https://doi.org/10.1109/ICDM.2009.128

Choi, W., Pantofaru, C., Savarese, S., 2011. Detecting and tracking people using an RGB-D camera via multiple detector fusion. IEEE Int. Conf. on Computer Vision Workshops, p.1076-1083. https://doi.org/10.1109/ICCVW.2011.6130370

Comaniciu, D., Ramesh, V., Meer, P., 2003. Kernel-based object tracking. *IEEE Trans. Patt. Anal. Mach. Intell.*, **25**(5):564-577. https://doi.org/10.1109/TPAMI.2003.1195991

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, p.886-893. https://doi.org/10.1109/CVPR.2005.177

Donoho, D.L., 2006. Compressed sensing. *IEEE Trans. Inform. Theory*, **52**(4):1289-1306. https://doi.org/10.1109/TIT.2006.871582

Hong, Z.B., Mei, X., Prokhorov, D., *et al.*, 2013. Tracking via robust multi-task multi-view joint sparse representation. IEEE Int. Conf. on Computer Vision, p.649-656. https://doi.org/10.1109/ICCV.2013.86

Lan, X.Y., Ma, A., Yuen, P., 2014. Multi-cue visual tracking using robust feature-level fusion based on joint sparse representation. IEEE Int. Conf. on Computer Vision and Pattern Recognition, p.1194-1201. https://doi.org/10.1109/CVPR.2014.156

Ling, H.B., Bai, L., Blasch, E., *et al.*, 2010. Robust infrared vehicle tracking across target pose change using L1 regularization. IEEE Conf. on Information Fusion, p.1-8. https://doi.org/10.1109/ICIF.2010.5711902

Liu, B.Y., Yang, L., Huang, J.Z., *et al.*, 2010. Robust and fast collaborative tracking with two stage sparse optimization. European Conf. on Computer Vision, p.624-637. https://doi.org/10.1007/978-3-642-15561-1_45

Luber, M., Spinello, L., Arras, K.O., 2011. People tracking in RGB-D data with on-line boosted target models. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, p.3844-3849. https://doi.org/10.1109/IROS.2011.6095075

Ma, Z.A., Xiang, Z.Y., 2015. Robust visual tracking via binocular multi-task multi-view joint sparse representation. SAI Intelligent Systems Conf., p.714-722. https://doi.org/10.1109/IntelliSys.2015.7361219

Mei, X., Ling, H.B., 2009. Robust visual tracking using ℓ_1 minimization. IEEE 12th Int. Conf. on Computer Vision, p.1436-1443. https://doi.org/10.1109/ICCV.2009.5459292

Mei, X., Ling, H.B., 2011. Robust visual tracking and vehicle classification via sparse representation. *IEEE Trans. Patt. Anal. Mach. Intell.*, **33**(11):2259-2272. https://doi.org/10.1109/TPAMI.2011.66

Mei, X., Ling, H.B., Wu, Y., *et al.*, 2011. Minimum error bounded efficient ℓ_1 tracker with occlusion detection. IEEE Conf. on Computer Vision and Pattern Recognition, p.1257-1264. https://doi.org/10.1109/CVPR.2011.5995421

Ojala, T., Pietikäinen, M., Mäenpää, T., 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Patt. Anal. Mach. Intell.*, **24**(7):971-987. https://doi.org/10.1109/TPAMI.2002.1017623

Pei, S.C., Lin, C.N., 1995. Image normalization for pattern recognition. *Image Vis. Comput.*, **13**(10):711-723. https://doi.org/10.1016/0262-8856(95)98753-G

Porikli, F., Tuzel, O., Meer, P., 2006. Covariance tracking using model update based on Lie algebra. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, p.728-735. https://doi.org/10.1109/CVPR.2006.94

Ross, D.A., Lim, J., Lin, R.S., *et al.*, 2008. Incremental learning for robust visual tracking. *Int. J. Comput. Vis.*, **77**(1-3):125-141. https://doi.org/10.1007/s11263-007-0075-7

Song, S.R., Xiao, J.X., 2013. Tracking revisited using RGBD camera: unified benchmark and baselines. IEEE Int. Conf. on Computer Vision, p.233-240. https://doi.org/10.1109/ICCV.2013.36

Williams, O., Blake, A., Cipolla, R., 2005. Sparse Bayesian learning for efficient visual tracking. *IEEE Trans. Patt. Anal. Mach. Intell.*, **27**(8):1292-1304. https://doi.org/10.1109/TPAMI.2005.167

Wright, J., Yang, A.Y., Ganesh, A., *et al.*, 2009. Robust face recognition via sparse representation. *IEEE Trans. Patt. Anal. Mach. Intell.*, **31**(2):210-227. https://doi.org/10.1109/TPAMI.2008.79

Wu, Y., Lim, J., Yang, M.H., 2013. Online object tracking: a benchmark. IEEE Conf. on Computer Vision and Pattern Recognition, p.2411-2418. https://doi.org/10.1109/CVPR.2013.312

Yang, M., Zhang, L., 2010. Gabor feature-based sparse representation for face recognition with Gabor occlusion dictionary. European Conf. on Computer Vision, p.448-461. https://doi.org/10.1007/978-3-642-15567-3_33

Yilmaz, A., Javed, O., Shah, M., 2006. Object tracking: a survey. *ACM Comput. Surv.*, **38**(4):43-56. https://doi.org/10.1145/1177352.1177355

Yin, Z.Z., Collins, R.T., 2008. Object tracking and detection after occlusion via numerical hybrid local and global mode-seeking. IEEE Conf. on Computer Vision and Pattern Recognition, p.1-8. https://doi.org/10.1109/CVPR.2008.4587542

Zhang, K., Zhang, L., Yang, M.H., 2012. Real-time compressive tracking. European Conf. on Computer Vision, p.864-877. https://doi.org/10.1007/978-3-642-33712-3_62

Zhang, T.Z., Ghanem, B., Liu, S., *et al.*, 2012a. Low-rank sparse learning for robust visual tracking. European Conf. on Computer Vision, p.470-484. https://doi.org/10.1007/978-3-642-33783-3_34

Zhang, T.Z., Ghanem, B., Liu, S., *et al.*, 2012b. Robust visual tracking via multi-task sparse learning. IEEE Conf. on Computer Vision and Pattern Recognition, p.2042-2049. https://doi.org/10.1109/CVPR.2012.6247908

Zhang, T.Z., Ghanem, B., Liu, S., *et al.*, 2013. Robust visual tracking via structured multi-task sparse learning. *Int. J. Comput. Vis.*, **101**(2):367-383. https://doi.org/10.1007/s11263-012-0582-z

Zhang, T.Z., Liu, S., Ahuja, N., *et al*, 2015a. Robust visual tracking via consistent low-rank sparse learning. *Int. J. Comput. Vis.*, **111**(2):171-190. https://doi.org/10.1007/s11263-014-0738-0

Zhang, T.Z., Liu, S., Xu, C.S., *et al.*, 2015b. Structural sparse tracking. IEEE Conf. on Computer Vision and Pattern Recognition, p.150-158. https://doi.org/10.1109/CVPR.2015.7298610

Zhang, Z.Y., 1994. Iterative point matching for registration of free-form curves and surfaces. *Int. J. Comput. Vis.*, **13**(2): 119-152. https://doi.org/10.1007/BF01427149