Robust Superpixel Tracking via Depth Fusion

Yuan Yuan, *Senior Member, IEEE,* Jianwu Fang, and Qi Wang

***Abstract*—Although numerous trackers have been designed to adapt to the nonstationary image streams that change over time, it remains a challenging task to facilitate a tracker to accurately distinguish the target from the background in every frame. This paper proposes a robust superpixel-based tracker via depth fusion, which exploits the adequate structural information and great flexibility of mid-level features captured by superpixels, as well as the depth-map's discriminative ability for the target and background separation. By introducing graph-regularized sparse coding into the appearance model, the local geometrical structure of data is considered, and the resulting appearance model has a more powerful discriminative ability. Meanwhile, the similarity of the target superpixels' neighborhoods in two adjacent frames is also incorporated into the refinement of the target estimation, which helps a more accurate localization. Most importantly, the depth cue is fused into the superpixel-based target estimation so as to tackle the cluttered background with similar appearance to the target. To evaluate the effectiveness of the proposed tracker, four video sequences of different challenging situations are contributed by the authors. The comparison results demonstrate that the proposed tracker has more robust and accurate performance than seven ones representing the state-of-the-art.**

***Index Terms*—Computer vision, depth fusion, graph regularized sparse coding, object tracking, segmentation, superpixel.**

I. Introduction

THE ABILITY to design a robust object tracker has successfully established many applications in a wide range of fields, ranging from traffic surveillance [1], [2], human activity analysis [3], [4], and human–computer interaction [5]. Although numerous trackers [6]–[11] have been proposed with significant success, designing a tracker that can handle the challenges, such as large variation of shape and illumination, drastic pose change, small target and cluttered scene (e.g., similar target with background), is still a difficult task.

Existing trackers commonly exploit the cues from low-level visual ones to high-level structural ones. The reason for exploiting this information is to construct adaptive and discriminative appearance models for the target and background separation. As mentioned in [16] and [17], although low-level cues, such as color, gradient and texture, are effective for feature tracking and scene analysis [18], they are less effective in the context of object tracking because of their weak ability of the context exploitation for the target. For example, the ensemble tracker [19] differentiates a target from background as a pixel-based binary classification problem, which utilizes an eight-bin local orientation histogram calculated on the RGB channels. However, the pixel-based representation is rather limited in handling heavy background clutter. Kwon and Lee [20] model the target appearance by a group of patches, where the smoothness and steepness of the patches in color are considered as the main criterion for patch selection. While this tracker demonstrates a robust performance for nonrigid objects, the smoothness and steepness of the patches are vulnerable to the heavy illumination change and background clutter. The Haar-feature based trackers, such as online AdaBoost (OAB) tracker [13] and multiple instances learning (MIL) tracker [6], have the superior ability to handle severe appearance and illumination change. However, these trackers are designed for rigid object tracking, and have poor adaptation for drastic shape deformation and pose change. As for the high-level cues based trackers, they commonly explore the semantic knowledge of the object, such as contour information of the target [21], [22]. But the high-level clues themselves are difficult to extract, which remains a question to be solved. For example, Yin *et al.* [22] address the problem of tracking contour by involving subspace and a contour template. However, the exact contour is difficult to be extracted in cluttered background, especially for the small target.

Considering the drawbacks of the low-level and high-level cues, mid-level visual cues with more sufficient structure information may provide a trade-off. At this level, superpixel has been established as the most promising representation [12], [23]. In [12], a tracker that constructs a superpixel-based discriminative appearance model for the separation of target and background is proposed. By contributing a confidence map for every superpixel, the superpixels with the accepted confidence are determined as parts of the target. While this tracker demonstrates a convincing tracking performance for large shape distortion and pose change in their project, it is still vulnerable to the cluttered background. For example, if the background has a similar appearance to the target, the tracking results would generate drift. In addition, because of the cluttered background, the estimated confidence map for the small target may contain several possible candidate positions, and lead to a false target estimation.

Besides, most existing trackers only exploit the cues extracted from RGB channels, which do not exploit the spatial depth information of the scene. However, the depth cue,

Manuscript received March 6, 2013; revised May 24, 2013; accepted June 3, 2013. Date of publication July 16, 2013; date of current version January 3, 2014. This work was supported in part by the National Basic Research Program of China (Youth 973 Program) under Grant 2013CB336500, in part by the National Natural Science Foundation of China under Grants 61172143, 61379094 and 61105012, and in part by the Natural Science Foundation Research Project of Shaanxi Province under Grant 2012JM8024. This paper was recommended by Associate Editor H. Wang.

The authors are with the Center for OPTical IMagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, Shaanxi 710119, China (e-mail: yuany@opt.ac.cn; fangjianwu@opt.ac.cn; crabwq@opt.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2013.2273631

reflecting the distance between the scene and camera, has robust ability to distinguish the background with the target, even in the situation that traditional trackers cannot tackle. As mentioned in the visual tracking review [24], one of the main future directions of visual tracking is the combination of different features, such as contour and texture or multimodal sensory data. Therefore, fusing depth information is one novel attempt to the development of visual tracking.

For effectively adapting to the target shape and pose variation, small target and cluttered background, this paper proposes a new superpixel-based tracker by fusing image depth information. Different from the pixel-based fusion strategies [25], [26], this paper aims to fuse the depth map with the superpixel-based target estimation, which has not been exploited in previous literature. With the flexible structure of superpixel, the shape and pose variation can be handled. Meanwhile, different depths of the target and background provide a robust target extraction from cluttered background. Because of these superiorities, more robust tracking results can be obtained, as shown in Fig. 1.

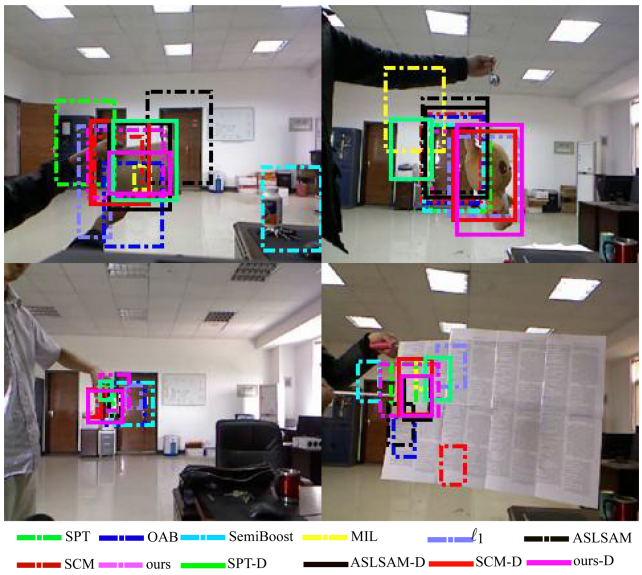


Fig. 1. Typical comparison results of the proposed tracker and seven competitive ones. The demonstrated results are generated by the proposed tracker with (ours-D) and without depth fusion (ours), SPT [12], OAB [13], SemiBoost [14], MIL [6], ℓ_1 tracker [8], SCM tracker [9] and ASLSAM tracker [15]. Besides, three trackers are selected to incorporate depth clue to be compared with the proposed one. They are respectively denoted as SPT-D, ASLSAM-D, SCM-D. The results show that existing trackers without fusing the depth cue fail to tackle the severe shape or pose variation, small target whose width is only several pixels, and the cluttered background with the similar scene to the target. The three trackers including depth clue are also inferior to the proposed one.

A. Overview of the Proposed Tracker

The flow chart of the proposed tracker is illustrated in Fig. 2. From the figure, it can be seen that the tracker is divided into three parts: superpixel-based target estimation in the RGB channels, target depth association in the depth channel, and fusion of the estimated target regions from the two sources.

For the superpixel-based target estimation, it is formulated as computing a confidence map for the target searching region. The size of the searching region is defined as twice the size of the estimated target in the previous frame. Then, the region is segmented into numerous superpixels. Because the number of the target superpixels only takes up a small part of the searching region. The target estimation problem can be formulated as a sparse optimal problem that each superpixel can be sparsely represented by a set of basic elements. To this end, this paper constructs a sparse superpixel-based discriminative appearance model (SSDAM), which introduces graph-regularized sparse coding (GrapghSC) for the first time to estimate the target. With the SSDAM, a coarse confidence map is computed. In order to refine it, the similarity of target superpixels' neighborhoods for two adjacent frames is considered. In addition, to adapt to the variation caused by the target and background, the SSDAM is constantly updated.

In terms of the target depth association, it aims to find the optimal target depth region by matching the candidate depth region with the predefined target depth model. For this purpose, an efficient graph segmentation [27] is introduced to segment the examined searching region into several smaller ones. Then, each candidate depth region is compared with the template by depth histogram, area and shape prior constraints. Through this process, the most similar one with the target model is determined as the target region.

With the estimated target region in RGB channels and depth channel, the fusion strategy follows an assumption that if different cues occur simultaneously in an estimated target region, then the region belongs to the target. Therefore, a $\sim \mathbb{XOR}$ operator is utilized to achieve the fusion process.

B. Contributions

The main contributions of the proposed tracker are as follows.

First, this paper proposes a SSDAM for the target estimation. From the review on sparsity-based trackers [28], most existing appearance models assume that an element in the dictionary is independent of all others. Obviously, this assumption ignores the correlation among the basic elements, and may introduce more potential noises when target estimation is performed. Therefore, this paper introduces a GraphSC [29] to learn the superpixels' sparse representation, which uses a graph Laplacian matrix as a smooth operator to consider the local geometrical structure of data, and achieves a more powerful discriminative ability. This is different from [30], which considers the geometrical structure of the target and templates, but needs more templates to sparsely represent the target in the tracking process, and has a higher computational cost.

Second, it is the first attempt to fuse the depth cue with superpixel-based target estimation. Traditional pixel-based fusion strategies need pixel-to-pixel registration between the images from the depth and RGB channels. The fusion strategy applied in this paper is different in that it works at the region-level. The only requirement for the later is that the centers of the target at different images should be generally near each other. To realize this requirement, the depth and RGB images are roughly registered and a voting strategy is then adopted.

Last, several video sequences representing different challenging situations are captured and labeled as the benchmark

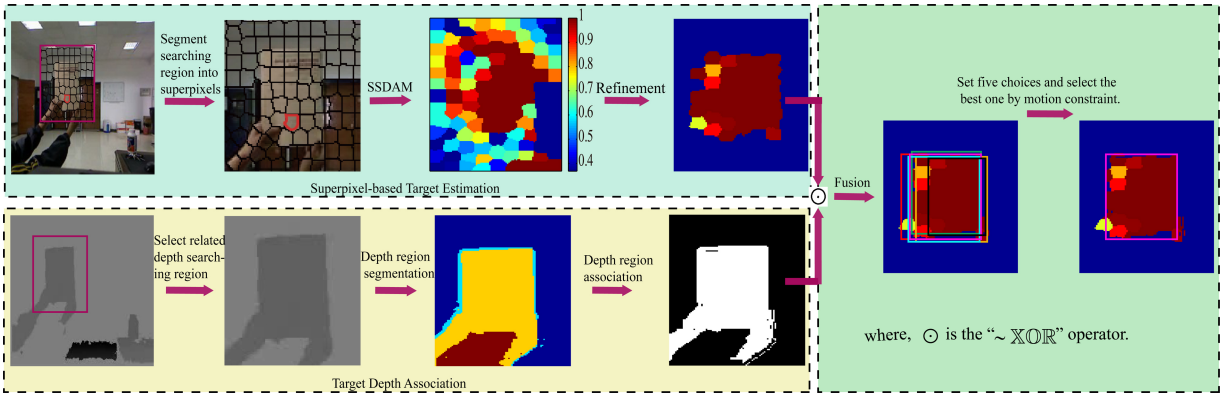


Fig. 2. Flow chart of the proposed tracker.

for the evaluation. Although there are many publicly available data sets for tracking research [6], [12], [31], they only have RGB channels. Different from these data sets, the sequences contributed in this paper have both the RGB and depth channels for every frame.

The remainder of this paper is organized as follows. Section II reviews the related work. Section III presents the proposed SSDAM model in detail. Section IV elaborates the target depth region association, and the depth fusion strategy is described in Section V. Section VI reports all the experimental results to prove the effectiveness of the proposed tracker, followed by some discussions. Section VII gives the conclusion in the end.

II. Related Works

Since the proposed tracker in this paper concentrates on the fusion between depth cue and superpixel-based appearance model, we restrict the literature review to the trackers that incorporate depth information. From the investigation of the trackers fusing depth map, they can be classified into two categories according to the difference of depth map extraction: passive-based and active-based. The depth maps from the passive-based trackers are explicitly recovered from the multiview images or image sequences, while the depth maps from the active-based ones are obtained by emitting kinds of optical lights to measure the distance between the scene and the camera, one typical example of which is the Kinect sensor.

For the first category, there are many examples [25], [32]–[34]. Most of these methods recover the depth map by multiview geometry, which needs to conduct camera collaboration and correspondence, leading to a high computational cost. For example, Michel *et al.* [32] proposed a monocular model-based 3-D object tracker. The initial step is to conduct the camera calibration and define an object model. Then, they update the object state at current time by matching the extracted edges and nodes of the object with the predefined object models. A recent work for multiple person tracking proposed in [25] also conducted the human tracking from multiview videos. The strategy for the target extraction is based on segmentation for estimated depth map and RGB visual image. Then, they refine the segmentation result by motion compensation and uncertainty refinement.

As for the second category, a typical example is the Kinect sensor. Since this kind of sensor can capture the depth map in real time, many trackers have utilized them [26], [35]–[38]. For example, several hand trackers [35]–[37] utilized Kinect sensor to generate the depth map. They conduct tracking by extracting the hand contour via fusing skin color and hand's depth. García *et al.* [26] extended the condensation algorithm to a RGB-D tracker for arbitrary objects. With the depth extracted from the Kinect sensor, they train a background/target classifier, which is boosted from the feature pool of grayscale, color, and depth. Fanelli *et al.* [38] presented a 3-D head tracker to estimate the head pose with Kinect, and extended the regression forests algorithm to classify the head pose. Teichman and Thrun [39] described a 3-D tracker based on semi-supervised learning, whose depth is generated from the LIDAR depth sensor. They initialize a classifier by the segmented object in the first frame, and then iteratively retrain the classifier with the new segmented object.

The aforementioned trackers fusing depth map are all within the pixel-wise level. Meanwhile, the fusion process commonly needs pixel-to-pixel registration among images from depth and RGB channels. With the above literature review, the proposed tracker in this paper is presented as follows.

III. Sparse Superpixel-based Discriminative Appearance Model (SSDAM)

In this section, the proposed SSDAM is presented in detail. First, in order to construct the SSDAM, the superpixel' feature representation is described. Then, we draw forth the SSDAM to compute a coarse confidence map for candidate superpixels. Based on the assumption that the related target's superpixels have similar neighborhoods for two adjacent frames, the similarity preservation constraint is utilized to refine the coarse confidence map.

A. Feature Representation for Superpixel

The superpixels in this paper are extracted by the simple linear iterative clustering (SLIC) [40] segmentation method, which is simple and has fairly low computational cost. After that, the photometric and geometrical characteristics are exploited for its representation.

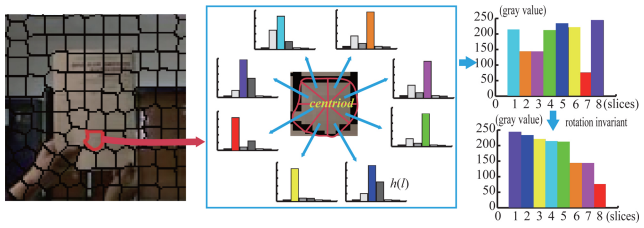


Fig. 3. Pie-structure-based histograms for superpixel.

Denote u as a superpixel in the RGB image and $\mathbf{m}_u \in \mathbb{R}^{1\times 3}$ and $\mathbf{v}_u \in \mathbb{R}^{1\times 3}$ its mean and variance, respectively. Let a_u be the area, and $\mathbf{c}_u$=(x_u, y_u) its centroid of u. To describe the geometrical distribution of u, the superpixel is parsed into L pie slices as demonstrated in Fig. 3, each of which begins at (x_u, y_u). For each slice, it has a gray histogram. We select the maximum component $h_u(l)$ of each histogram to represent the superpixel's distributed gray characteristic, $\mathbf{H}_u = \{h_u(l)|l = 1, ..., L\}$. The definition for this histogram structure is inspired by [40]. But this representation is sensitive to image rotation. To make rotation invariant, $\mathbf{H}_u$ is sorted according to the L values from large to small. Furthermore, the entropy of the normalized histogram is also calculated to describe superpixel's texture. To be specific, it is defined as

$$e_u = -\sum_{l=1}^{L} h_u(l) \log h_u(l). \tag{1}$$

With the above formulation, the feature vector of superpixel u is defined as $\mathcal{F}_u = \{\mathbf{P}_u,\mathbf{G}_u\}$. It comprises the following two components, the photometric one $\mathbf{P}_u$=$\{\mathbf{m}_u,\mathbf{v}_u,e_u\}$ and the geometrical one $\mathbf{G}_u$=$\{a_u,\mathbf{c}_u,\mathbf{H}_u\}$, respectively.

As for the photometric and geometrical cues, they are commonly combined as a vector. However, there is no spatial correlation between them. Meanwhile, according to [42], it suggests that different cues should be utilized separately. It follows an assumption that if one kind of cue fails, another one can be supplementary. Therefore, in this paper, the photometric cues are utilized to construct SSDAM. For a supplement and refinement, the geometrical cues are utilized in the subsequent processing.

B. Sparse Superpixel-Based Discriminative Appearance Model

Consider the group of candidate superpixels segmented from the examined frame. We can estimate each superpixel's label indicating the target or background. In this paper, the target estimation is formulated as a sparse optimal problem. We suppose each superpixel can be expressed on a set of basic elements. Therefore, this paper proposes a SSDAM, which, for the first time, embeds a GraphSC [29] to take account for the data's geometrical structure. In the following description, first, the motivation of utilizing GraphSC is presented, and then our SSDAM is introduced.

1) *Motivation of Utilizing GraphSC:* In recent years, sparse coding has been established as a promising tool for problem solving in computer vision. The assumption of sparse coding is that the original data $\mathbf{y} \in \mathbb{R}^{d\times 1}$ can be encoded by the sparse linear combination of N basic elements

$$\min_{\mathbf{a}} \|\mathbf{y} - \mathbf{D}\mathbf{a}\|_F^2 + \lambda\|\mathbf{a}\|_1 \tag{2}$$

where $\mathbf{D} \in \mathbb{R}^{d\times N}$ is the encoding dictionary, $\mathbf{a} \in \mathbb{R}^{N\times 1}$ specifies the encoding coefficients, and λ is the Lagrangian multiplier, which balances the importance of the sparseness and the reconstruction error. In fact, (2) is known as the ℓ_F-norm form in Lasso, and is exploited in sparsity-based trackers [8], [9] proposed recently. Although Lasso enjoys great performance, it is worth noting that this method implicitly assumes that an element in the dictionary is independent of all others. Apparently, it ignores the correlation among the basic elements, which may introduce more potential noises when the target estimation is performed. For addressing it, this paper follows a manifold assumption inspired by [29], that if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are close to each other in the intrinsic geometry, the two data's representations $\mathbf{a}_i$ and $\mathbf{a}_j$ are also similar to each other. Based on this assumption, a Laplacian regularizer is incorporated into the original sparse coding, which exploits the local geometrical structure of data. Therefore, the motivation of utilizing GraphSC is to integrate correlations among candidate basic elements (superpixels). With the motivation of utilizing GraphSC, the detailed fumulation of SSDAM is presented as follows.

2) *SSDAM:* In this paper, the target searching region is a rectangle centered at the target location of the previous frame, twice its size in width and height. Then, it is segmented into m superpixels $\mathbf{Y}^p = [\mathbf{y}_1^p, \mathbf{y}_2^p, ..., \mathbf{y}_m^p]$, where $\mathbf{y}_j^p = [\mathbf{m}_j,\mathbf{v}_j,e_j]^T$ is a column vector, and is represented by the photometric cues. These superpixels construct a nearest neighbor graph $\mathcal{G}$ with m vertices, where each vertex represents a superpixel. Considering the correlations among superpixels, let $\mathbf{W}$ be a weight matrix of $\mathcal{G}$ with $W_{ji} = 1$ representing two superpixels $\mathbf{y}_j^p$ and $\mathbf{y}_i^p$ are among their related k-nearest neighbors, and vice versa. Then, a diagonal matrix $\mathbf{C}$ is defined to represent the importance of $\mathbf{y}_j^p$, where $C_{jj} = \sum_{i=1}^{m} W_{ji}$.

With the weight matrix $\mathbf{W}$ of graph $\mathcal{G}$, we need to map the m superpixels to their related k-nearest neighbors, and obtain a sparse representation coefficient matrix $\mathbf{A} = [\mathbf{a}_1^T, \mathbf{a}_2^T, ..., \mathbf{a}_m^T]$, where $\mathbf{a}_m^T$ specifies the encoding coefficient vector of the mth superpixel. To realize it, the following objective function is chosen:

$$\frac{1}{2}\sum_{j=1}^{m}\sum_{i=1}^{m}\|\mathbf{a}_j - \mathbf{a}_i\|^2 W_{ji} = \mathrm{Tr}(\mathbf{A}\mathbf{L}\mathbf{A}^{\mathbf{T}}) \tag{3}$$

where $\mathbf{L} = \mathbf{C} - \mathbf{W}$ is the Laplacian matrix. To minimize this equation is to find an optimal solution for $\mathbf{a}_i$, which is given by the eigenvector with the smallest nonzero eigenvalue. In other words, it guarantees that the connected superpixels of the graph $\mathcal{G}$ stay as close to each other as possible. By integrating $\mathbf{L}$ into the traditional sparse coding, our objective function is defined as

$$\begin{aligned} &\min_{\mathbf{A}} \|\mathbf{Y}^p - \mathbf{D}\mathbf{A}\|_F^2 + \lambda_1 \mathrm{Tr}(\mathbf{A}\mathbf{L}\mathbf{A}^T) + \lambda_2 \sum_{i=1}^{m} \|\mathbf{a}_j\|_1, \\ &\text{s.t. } \|\mathbf{d}_j\| < c,\ j = 1, ..., k \end{aligned} \tag{4}$$

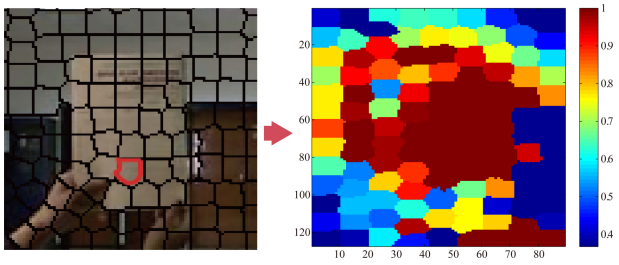


Fig. 4. Confidence map generated by SSDAM. The left one is the original segmented RGB image and the right one is the obtained confidence map.

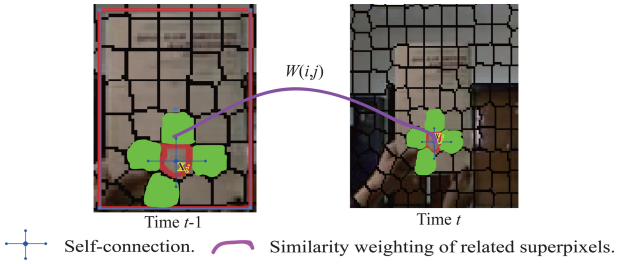


Fig. 5. Similarity measurement for two corresponding superpixels in adjacent frames.

where $\mathbf{d}_j$ represents the basis vector of $\mathbf{D}$, and c is the radius of the k-nearest neighbor. λ_1 is the coefficient of the regularizer.

In this paper, the dictionary $\mathbf{D}$ is generated by k-means from clustering the superpixels in the target region. For an adaption to the variation of the target and background, the dictionary $\mathbf{D}$ is constantly updated for every ten frames. Assume the latest updating time to be t_{tr}, and the number of superpixels belonging to the target at t_{tr} to be d. Then, the number of cluster center k is heuristically set as $d/6$.

In terms of solving $\mathbf{A}$, (4) with ℓ_1-regularization is non-differentiable when $\mathbf{a}_j$ contains 0. Therefore, an optimization method based on coordinate descent introduced in [29] is utilized in this paper, which updates $\mathbf{a}_j$ individually while holding all the other vectors constant. Thus, (4) is rewritten as

$$\min_{\mathbf{a}_j} \sum_{j=1}^{m} \|\mathbf{y}_j^p - \mathbf{D}\mathbf{a}_j\|_2^2 + \lambda_1 \sum_{i,j=1}^{m} L_{ij}\mathbf{a}_i^T\mathbf{a}_j + \lambda_2 \sum_{i=1}^{m} \|\mathbf{a}_j\|_1. \tag{5}$$

Actually, solving (5) is equivalent to solving the following equation:

$$\begin{aligned} &\min_{\mathbf{a}_j} \|\mathbf{y}_j^p - \mathbf{D}\mathbf{a}_j\|_2^2 + \lambda_1 L_{ji}\mathbf{a}_i^T\mathbf{a}_j + \lambda_2 \|\mathbf{a}_j\|_1, \\ &L_{ji}\mathbf{a}_j^T\mathbf{a}_j = \lambda_1 L_{jj}\mathbf{a}_i^T\mathbf{a}_j + 2\lambda_1 \mathbf{a}_i^T \sum_{j\neq i} L_{ji}\mathbf{a}_j. \end{aligned} \tag{6}$$

This can be solved by the feature-sign search algorithm [43], which is rather slow for pixel-based sparse coding. As in this paper, the image is represented as superpixels and the optimization process directly searches the optimal target representation among the superpixels. Therefore, the process is fast and adequate for our object tracking.

After constructing the SSDAM, the confidence Cp_j^t of the superpixel $\mathbf{y}_j$ in the searching region at time t is estimated as

$$Cp_j^t = e^{\left(-\frac{\|\mathbf{y}_j - \mathbf{D}\mathbf{a}_j\|_2^2}{\sigma}\right)} \tag{7}$$

where the variable σ is fixed to a constant that normalizes the superpixel's construction error. All of the confidences of superpixels, which belong to the searching region, construct a confidence map $\mathcal{C}_{ssdam}^t$, where $Cp_j^t \in \mathcal{C}_{ssdam}^t$.

From the computed confidence map as shown in Fig. 4, the estimated target region may introduce some disturbing superpixels, which is caused by the similar data structure in the searching region. In order to refine it, this paper proposes a strategy modeled by the similarity preservation of superpixels' neighborhoods in two adjacent frames.

C. Similarity Preservation of Superpixels' Neighborhoods

In this subsection, we propose a similarity preservation strategy to refine the coarse confidence map obtained by SSDAM. Based on the work in [44], the similarity measure depends on not only two individual samples but also their corresponding contexts. Our principle is that not only the superpixels of the target in adjacent frames should have a similar characteristic, but also their neighborhoods should resemble each other. This assumption holds the same view as the work in [44]. To realize this principle, the connection between the superpixel itself and its local neighborhoods is defined first, which is called self-connection as shown in Fig. 5. Then, the similarity between superpixels in two adjacent frames is computed. With the calculated similarity, the most reliable superpixels are maintained and the less confident ones are discarded. Finally, the refined confidence map contains almost only the target superpixels with high confidences.

1) *Definition of Self-Connection:* Denote $\mathbf{x}_i^g$ to be one of the target's superpixels at time $t-1$, and $\mathbf{y}_j^g$ to be one of the superpixels belonging to the searching region at time t. Every superpixel here is represented by its geometrical cues. The connection between the superpixel $\mathbf{x}_i^g$ and its 4-neighborhoods $\{\mathbf{x}_p^g\}_{p=1}^4$ is defined as the distance between the $\mathbf{H}_i$ and each $\mathbf{H}_p$. Here, the $\mathcal{X}^2$ test statistic is selected as the distance metric

$$\rho_{i,p} = \frac{1}{L}\sum_{l}^{L} \frac{(h_i(l) - h_p(l))^2}{h_i(l) + h_p(l)} \tag{8}$$

where L is the pie slice number of the superpixel. Similarly, the connection between the superpixel $\mathbf{y}_j^g$ and its neighborhoods $\{\mathbf{y}_q^g\}_{q=1}^4$ at time t is defined as $\rho_{j,q}$.

2) *Similarity of the Target's Superpixels:* To weigh the similarity of the corresponding superpixels of the target, their appearances are compared with each other, in addition to their self-connections. We also take $\mathbf{x}_i^g$ and $\mathbf{y}_j^g$ as an example. The matching cost is set as

$$\begin{aligned} W(i,j) &= \omega(i,j) + f_{i,j} \\ \omega(i,j) &= \left|\frac{a_i}{a_j} - 1\right| + \rho_{i,j} + \sum_{p=1,q=1}^{4} \left(\left|\frac{a_p}{a_i} - \frac{a_q}{a_j}\right| + |\rho_{i,p} - \rho_{j,q}|\right) \\ f_{i,j} &= \frac{\|\mathbf{c}_i - \mathbf{c}_j\|_2}{r} \end{aligned} \tag{9}$$

where $\omega(i,j)$ indicates the correlation of superpixels' areas and gray scale distributions in two adjacent frames. $f_{i,j}$ specifies the displacement of superpixels' centers in two adjacent frames

and the normalizing factor r is set to 20 in all experiments. The similarity degree of $\mathbf{x}_i^g$ and $\mathbf{y}_j^g$ is then defined as $e^{-W(i,j)}$.

Based on the above formulation, the confidence of superpixel $\mathbf{y}_j^g$ is recomputed as

$$Cp_j^t = \begin{cases} Cp_j^t & \text{if } e^{-W(i,j)} > T(i,j) \\ 0 & \text{if } e^{-W(i,j)} < T(i,j) \end{cases} \tag{10}$$

where $T(i,j) = \max(\exp(-W(i,j))) - \sigma_f$ denotes a threshold that represents the similarity degree of the related superpixels, and σ_f denotes the standard deviation of $\exp(-W(\cdot))$. After refining Cp_j for each superpixel, a final confidence map C_t of the searching region can be obtained, where $Cp_j \in C_t$. In this map, the region of the target is generally maintained and the unrelated ones are mostly discarded.

IV. Target Depth Association

As mentioned before, this paper focuses on the fusion of the target appearance and depth cues. For the generation of the related depth map, Kinect sensor is employed, which was released by Microsoft on November 2010 and is very effective in generating real-time sequences.

After obtaining the depth sequence, we need to associate the candidate target depth with the predefined target depth model in the first frame. With respect to extracting the depth region at current time, an efficient graph segmentation [27] is employed to segment the searching region into several regions. With the previously defined depth model, the new target depth region is associated according to the regions' depth histogram, area and shape prior. In the following, the depth model is first defined and the strategy of the target depth association is then introduced.

A. Depth Model of Target

The target's model is represented by the depth distribution, area, and shape. For the depth distribution, it is denoted by a normalized depth histogram. In terms of the shape cue, this paper introduces a signed distance function inspired by [45]. To be specific, given a shape region $\Omega \subset \mathbf{R}^2$, denote x to be its centroid. The signed distance function is then defined as

$$\phi = \begin{cases} +1, x \in \Omega \\ -1, x \in \mathbf{R}^2 \backslash \Omega. \end{cases} \tag{11}$$

This strategy is illustrated in Fig. 6, where $Dcb \leftarrow Dcb \times \phi$ specifies the distance between the boundary and the centroid x. By that, a normalized shape histogram is obtained by counting the statistics of Dcb.

After the above definition, the target's model is denoted as three parts, depth histogram HD_o, area r_o, and shape histogram HS_o.

B. Association of Examined Depth Map

Assume the target's depth searching region at time t is segmented into N_t regions. Their characteristics are denoted by HD_t^n, r_t^n and HS_t^n, where $n = 1, ..., N_t$ is the index of the segmented depth region. The association constraints are then expressed as three-fold: target's depth histogram constraint, area constraint, and shape prior constraint.

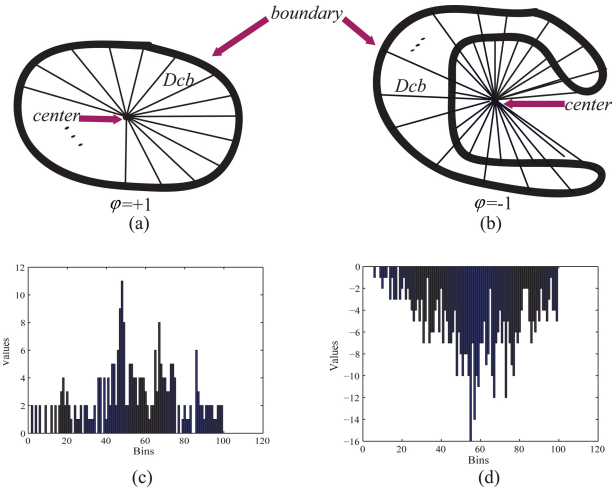


Fig. 6. Shape histogram. (a) Centroid of the region is inside of the region. (b) Centroid of the region is outside of the region. (c) Normalized shape histogram of (a). (d) Normalized shape histogram of (b). The bins' number of shape histograms is set as 100 for all experiments.

For the target's depth histogram constraint, it is defined by matching the depth histogram $\{HD_t^n\}_{n=1}^N$ at time t and HD_o with $\mathcal{X}^2$ test statistic

$$\begin{aligned} D_t^{HD}(o,n) &= \frac{1}{I_1} \sum_{i=1}^{I_1} \frac{(HD_o(i) - HD_t^n(i))^2}{HD_o(i) - HD_t^n(i)} \\ D_t^{HD^n} &= \exp(-\frac{D_t^{HD}(o,n)}{\max(D_t^{HD}(o,n))}) \end{aligned} \tag{12}$$

where $D_t^{HD}(o,n)$ represents the matching cost of HD_t^n and HD_o, $D_t^{HD^n}$ is the normalization of $D_t^{HD}(o,n)$, and I_1 is the number of bins in the target's depth histogram (set to 255).

With respect to the area constraint, the principle is that the target's area should be constant in a few successive frames. Therefore, the area distance is defined as $e^{-|(r_t^n/r_o)-1|}$.

In terms of the shape prior constraint, it is defined by matching the $\{HS_t^n\}_{n=1}^N$ and HS_o with modified $\mathcal{X}^2$ test

$$\begin{aligned} D_t^{HS}(o,n) &= \frac{1}{I_2} \sum_{i=1}^{I_2} \frac{(HS_o(i) - HS_t^n(i))^2}{|HS_o(i) - HS_t^n(i)|} \\ D_t^{HS^n} &= \exp(-\frac{D_t^{HS}(o,n)}{\max(D_t^{HS}(o,n))}) \end{aligned} \tag{13}$$

where $D_t^{HS}(o,n)$ represents the matching cost of HS_t^n and HS_o, $D_t^{HS^n}$ specifies the normalization of $D_t^{HS}(o,n)$ and I_2 is the number of bins for shape histogram (set to 100). The $|\cdot|$ operator could make $D_t^{HS}(o,n)$ avoid a negative value that may generate a false matching.

With the above distance constraints, the target's depth region at time t is estimated as $\mathcal{O}_t$, whose depth histogram, area, and shape histogram are subject to

$$\arg\max_n \{\gamma D_t^{r^n} D_t^{HD^n} + (1-\gamma) D_t^{HS^n}\} \tag{14}$$

where γ is set to 0.5, empirically.

With the depth association, all of the pixels falling into $\mathcal{O}_t$ is set to 1, and 0 vice versa.

V. Fusion of Target Estimation and Depth Association

With the estimated target and the associated depth region, the fusion process is conducted in this section. In this paper, we adopt a simple but efficient approach for fusing the confidence map $\mathcal{C}_t$ obtained by SSDAM in RGB channels and the associated depth region $\mathcal{O}_t$ in depth channel.

For the fusion of $\mathcal{C}_t$ and $\mathcal{O}_t$, we follow an assumption that if the superpixels belong to both $\mathcal{C}_t$ and $\mathcal{O}_t$, the superpixels belong to the target. Therefore, the fusion process can be formulated as

$$\mathcal{C}_t^{in} = \mathcal{C}_t \odot \mathcal{O}_t \tag{15}$$

where $\mathcal{C}_t^{in}$ represents the confidence map falling into $\mathcal{O}_t$, and $\odot$ is the $\sim \mathbb{XOR}$ operator. One example of the fusion result is shown in Fig. 2.

The obtained $\mathcal{C}_t^{in}$ covers most of the target region as illustrated in Fig. 2, which can provide a credible clue for further accurate localization. Based on $\mathcal{C}_t^{in}$, five candidate target states $\{\mathbf{X}_t^f\}_{f=1}^5$ are sampled. Each candidate target state is constructed by a group of superpixels whose confidences are the top K ones in $\mathcal{C}_t^{in}$. Assume the number of target's superpixels is estimated as K_{t-1} in previous time. For a robust selection of K, it is chosen as five values: $\{K_{t-1}-2,\ K_{t-1}-1,\ K_{t-1},\ K_{t-1}+1,\ K_{t-1}+2\}$.

For a robust localization of the target, the estimation of $\hat{\mathbf{X}}_t$ follows an assumption: The target state $\mathbf{X}_t$ to be predicted depends on not only the previous target state $\mathbf{X}_{t-1}$, but also the most recent target state $\mathbf{X}_{t_{tr}}$. Denote the target state at time t to be $\mathbf{X}_t = [x_t, y_t, w_t, h_t, s_t]$, where x_t and y_t denote the center of the target, w_t and h_t represent the width and height of the target, and s_t specifies the ratio of w_t and h_t.

The optimal target state at time t is determined by maximizing *a posteriori*

$$\hat{\mathbf{X}}_t = \arg\max_f p(\mathbf{X}_t^f|\mathbf{X}_{t-1})p(\mathbf{X}_t^f|\mathbf{X}_{t_{tr}}) \tag{16}$$

where $p(\mathbf{X}_t^f|\mathbf{X}_{t-1})$ is determined by the Gaussian distribution $\mathcal{N}(\mathbf{X}_t^f; \mathbf{X}_{t-1}, \Sigma_t)$, $p(\mathbf{X}_t^f|\mathbf{X}_{t_{tr}})$ is estimated by $\exp(-\frac{\|\mathbf{X}_t^f - \mathbf{X}_{t_{tr}}\|_2^2}{r_{t_{tr}}})$, and $r_{t_{tr}}$ is set to be a constant (set to 20 in all experiments) for balancing the importance of $\mathbf{X}_{t_{tr}}$ and $\mathbf{X}_t^f$. So far, the target localization is presented. The steps of the proposed tracker are summarized in Algorithm 1.

Algorithm 1 Proposed Tracker

Initialization:

Initialize all parameters in our tracker before tracking. All of these parameters are presented in experiments in detail.

Updating:

for $t = t_{tr}$

1: Segment the surrounding region of $\mathbf{X}_{t_{tr}}$ into $m_{t_{tr}}$ superpixels and extract their feature pool $\{\mathcal{F}_j\}_{j=1}^{m_{t_{tr}}}$.
2: Select the superpixels belong to the target at time t_{tr} to update the dictionary $\mathbf{D}$.

end

Tracking:

for $t = t_{tr}$ to $t_{tr} + m$ (m is set as 10 in this paper)

1: Segment the surrounding region of $\mathbf{X}_t$ in to m_t superpixels and extract their features. Compute the target confidence map $\mathcal{C}_{ssdam}$ using (7).
2: Refine the $\mathcal{C}_{ssdam}$ and obtain $\mathcal{C}_t$ by (10).
3: Segment the depth map of the target searching region at time t into N_t segments, and associate the target depth region$\mathcal{O}_t$ by (14).
4: Conduct the fusion by (15).
5: Sort the values of $\mathcal{C}_t^{in}$ in a descending order.
6: Select K superpixels with the top K values in $\mathcal{C}_t^{in}$, where K is denoted as five selections: $\{K_{t-1}-2,\ K_{t-1}-1,\ K_{t-1},\ K_{t-1}+1,\ K_{t-1}+2\}$.
7: Select the bounding box for the five group superpixel sets, and compute their related target state $\mathbf{X}_t^f$, $f = 1, ..., 5$.
8: Conduct the localization by (16).

end

Output: $\hat{\mathbf{X}}_t$.

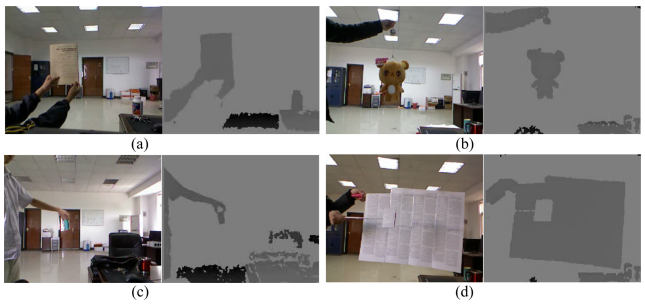


Fig. 7. Four challenging video sequences captured in this paper. (a) *Book*. (b) *Bear*. (c) *teaCan*. (d) *Paper*.

VI. Experiment Results

A. Data Sets

This paper aims to design a robust tracker via depth fusion. However, the publicly available data sets only have the RGB channels, and are not adequate for our experiments. In this paper, four challenging video sequences with depth channel captured by ourselves are contributed to prove the effectiveness of the proposed tracker. All of the sequences are recorded by the Kinect sensor. Typical frame shots of RGB channels and their corresponding depth channel are shown in Fig. 7. The challenges of these sequences contain large shape and illumination variation (*Book*), drastic pose change (*Bear*), small target (*teaCan*) and heavy cluttered background with the similar appearance to the object (*Paper*). The frame size of every video sequence is 195×240, and the frame numbers are 700, 233, 200 and 100, respectively.

B. Implementation Details

First, the parameter configuration in this paper is presented. Then, the experimental setups are designed for the trackers' performance analysis.

1) *Parameter Configuration:* In this paper, the superpixel generation is conducted by the SLIC [40] segmentation

TABLE I
PARAMETER CONFIGURATION FOR SUPERPIXEL'S GENERATION AND DEPTH REGION GENERATION

Superpixel			Depth segmentation		
Parameters	SI	SC	K	σ	$Minarea$
$book$	10.0	2.0	500	1	200
$bear$	10.0	2.0	500	1	200
$teaCan$	4.0	2.0	200	1	60
$paper$	10.0	2.0	500	1	200

method, which adapts a k-means clustering method to generate superpixels efficiently. In this method, there are two parameters to be set, which are the size of sample interval (SI) and the superpixel compactness (SC). The detailed parameters of superpixel generation are given in Table. I. Among them, the setting of SI is related to the target scale. For example, SI for *teaCan* sequence is apparently smaller than others. The reason is that the target needs enough number of superpixels to represent it. The parameter SC controls the superpixel's compactness. In this paper, the superpixels for all kinds of targets need to maintain compact. So SI is set the same for all sequences. For the number of pie slices L in (1), it is set to 8, which is adequate for the superpixel representation in this paper, especially for the small superpixels, which cannot be parsed into more than 8 pie slices.

In addition, for depth region generation in Section IV, efficient graph segmentation is introduced to segment depth region. The parameters for this algorithm are K, σ and *Minarea*, where K controls the difference between two segmented regions, σ the smoothing parameter controlling the connection between adjacent segments and *Minarea* the minimum area of the generated segments. Similarly, the detailed setting for all the parameters is shown in Table. I. Based on the physical meaning of each parameter, we know that K and *Minarea* are related to the target's scale. Therefore, for *teaCan* sequence, K and *Minarea* are smaller than others. As for σ, the connection between adjacent segments for all the sequences is the same.

In addition, for the generation of Laplacian matrix $\mathbf{L}$ in (3), the detailed configuration is set as follows. The neighbor mode for $\mathbf{L}$ is k-nearest neighbor, and the number of neighbors is set as 5. Meanwhile, the distance metric is chosen as *Cosine* metric, and the weighting mode is selected as *HeatKernel*. λ_1 and λ_2 in (6) are set as 0.01 and 0.05, respectively.

2) *Experimental Setups:* In order to prove the efficiency and accuracy of the proposed tracker, competitive ones representing the state-of-the-art are utilized to contribute the comparison with the proposed tracker. Among them, the most similar work to this paper is the superpixel tracking (SPT) [12]. In addition, since this paper utilizes sparse coding, trackers by sparsity-based collaborative model (SCM) [9], adaptive structural local sparse appearance model (ASLSAM) [15] and ℓ_1 minimization (ℓ_1) [8] are also involved in the comparison. In addition, three popular trackers, OAB [13], SemiBoost tracker (SemiBoost) [14], and MIL tracker [6], which are considered baselines in this field are also included in the comparison list. The codes of all these trackers can be downloaded from the authors' web sites.

In addition, in order to illustrate the importance of the depth cue, we also conduct experiments with (ours-D) and without (ours) depth fusion. Since the proposed tracker includes both the RGB channels and depth channel, we also want to compare it with the ones including the RGB-D channels. But unfortunately, there is not publicly available tracking code fusing depth information to the best of the authors' knowledge. Therefore, to be fairer, this paper selects three trackers (SPT, ASLSAM and SCM) that are most related to this paper, and fuses the depth cue into them, denoted as SPT-D, ASLSAM-D, and SCM-D. As for choosing SPT, the reason is that it exploits the superpixel representation. Besides, because ASLSAM and SCM represent the state-of-the-art for sparsity-based trackers, they are also selected to prove the efficiency of fusion for depth cues. The related parameters of every tracker are carefully adjusted and the best result of every tracker is selected from five runs. The detailed fusion strategies of SPT, ASLSAM and SCM are expressed as follows.

1) SPT-D: Combine the depth information into the appearance histogram of each superpixel originally modeled by the clues from the RGB channels.
2) ASLSAM-D: Design a tracking flow similar to the ASLSAM in depth channel, and select the target state with maximum confidence obtained from RGB and depth channels for every frame.
3) SCM-D: The fusion strategy is similar to ASLSAM-D.

For a clearer demonstration, all the trackers are analyzed together in the quantitative evaluation and qualitative analysis.

C. Tracking Performance Analysis

In this subsection, we first conduct the quantitative evaluation. The evaluation metrics of quantitative evaluation are center location error (CLE), precision with accepted bias (PAB), and average center location error (ACLE). The CLE represents the displacement (in pixels) between the generated center and the ground truth for each frame in a sequence. PAB describes the tracker's stability, which is represented as the ratio of frames whose center location errors are below a predefined threshold. ACLE specifies the average center location error of all the frames in every sequence. In this paper, CLE and PAB are presented in Figs. 8 and 9, and ACLE is shown in Table II. From CLE and PAB, the demonstrated results indicate that ours-D is superior to other trackers, especially for the *Bear*, *teaCan*, and *Paper* sequences. In addition, the depth cue can apparently improve the trackers' performance. In the following description, the qualitative analysis of the trackers for tackling the following challenges is presented in detail.

1) *Large Variation of Shape and Illumination:* The first row (*Book* sequence) in Fig. 10 shows that the *Book* has severe shape deformation, such as rotating, folding and unfolding. Besides, the book cover appears illumination varying with the fowards and backwards movements. From the results in Fig. 10, the OAB, SemiBoost, ℓ_1, SCM and ASLSAM trackers drift to the background area. The reason is that these trackers are not designed for nonrigid objects, and the bounding box of

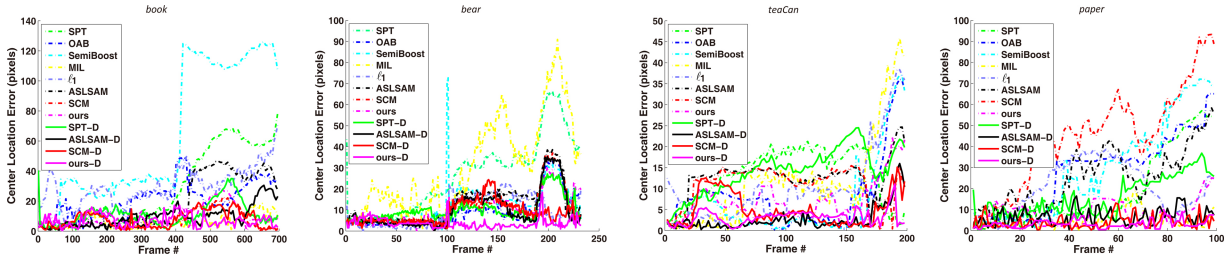


Fig. 8. Center location error. Horizontal axis is the frame index. Vertical axis is the location error (in pixels) between the target center and the ground truth. To be clearer, the trackers fused depth cue are all demonstrated by solid lines, and the trackers without fusing depth cue are represented by dashed lines.

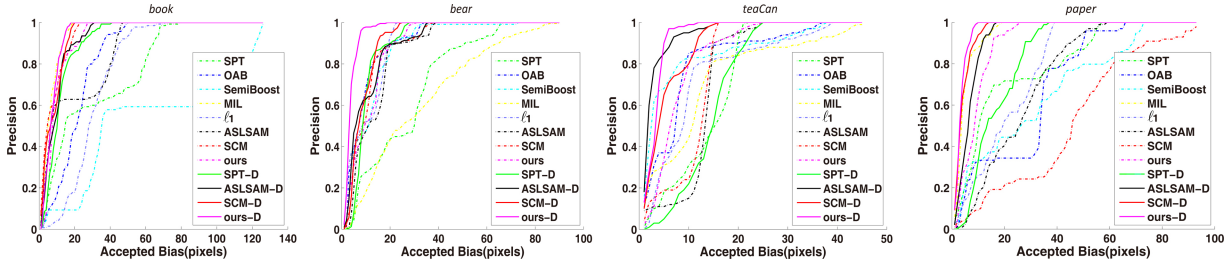


Fig. 9. Precision with accepted bias. Horizontal axis represents an accepted threshold bias (in pixels) between the target center and the ground truth center. Vertical axis is the ratio of the number of frames whose location error is below the threshold.

TABLE II

AVERAGE CENTER LOCATION ERROR (IN PIXELS). **BOLD** IN EACH ROW IS THE BEST CHOICE, *ITALIC* IS THE SECOND

	SPT	OAB	SemiBoost	MIL	ℓ_1	ASLSAM	SCM	ours	SPT-D	ASLSAM-D	SCM-D	ours-D
book	28.7	21.1	63.4	7.7	29.9	*7.3*	17.7	9.7	11.4	9.5	*7.3*	**6.0**
bear	26.3	9.3	11.8	31.0	9.9	10.6	13.1	10.2	10.2	9.9	*8.6*	**3.1**
teaCan	14.4	8.3	6.78	12.1	10.7	12.4	10.8	7.6	13.2	*3.2*	5.1	**2.7**
paper	18.9	27.9	29.2	4.9	20.4	44.6	25.3	8.9	16.3	6.5	*4.1*	**3.2**

the target is fixed and may introduce background interference. For the MIL tracker, although it shows an excellent location precision as shown in Figs. 8 and 9, the tracked bounding box shrinks to a small point, shown in the 464th frame. In other words, the scale adaptation of MIL tracker is inferior. In addition, SPT tracker is designed to tackle the shape deformation, but during tracking, the target introduces other cluttered background, such as the *Bookcase* in the 303th frame. Therefore, SPT also drifts to background area at last. As for our tracker without fusing depth cue, the tracking accuracy is a little weaker than the ASLSAM tracker representing the state-of-the-art. The main reason is that our SSDAM without background appearance model is vulnerable to the background clutter. With the depth map which can restrict the target at an approximately accurate location, the performance of SPT-D, ASLSAM-D and SCM-D improve obviously. However, ours-D demonstrates a superior performance for the shape deformation and varying illumination, especially for the scale adaptability.

2) *Drastic Pose Change:* The second row (*Bear* sequence) of Fig. 10 demonstrates that the *Bear* has a drastic pose variation. For the SPT, it is easy to be disturbed by the background clutter. Therefore, it shows a drift in early frames. Because the shape of the *Bear* is nonregular, the SCM and ASLSAM trackers, which initialize the target bounding box as a fixed rectangle, may introduce much background clutter, such as the *Door* from the 180th frame to the 230th frame. Another reason for the performances of SCM and ASLSAM is that the *Door* has the similar appearance cues to the *Bear*. Therefore, its template updating treats the template extracted from the *Door* region as positive. That is why drift occurs in SCM and ASLSAM trackers from the 180th frame to the 230th frame. For the OAB, SemiBoost and our tracker without fusing depth cue, they are influenced by the same issue as the one of SCM and ASLSAM. As for the MIL tracker, the holistic appearance model may put an inferior template into its positive bag. Therefore, the inferior template causes a severe drift. It seems that the ℓ_1 tracker demonstrates a good performance for the pose variation. However, the ℓ_1 tracker is influenced by the trivial background pixels more or less, shown in the 109th and 136th frame. This issue can also be verified in Fig. 9. After fusing depth cue, SPT-D, ASLSAM-D and SCM-D improve oneself obviously. Especially for the SCM-D, it can restrain the influence of *Door* region. However, its scale adaptability is a little weaker than ours-D. All in all, the proposed tracker is superior to the others.

3) *Small Target:* The object *teacan* in the *teaCan* sequence is rather small. By this sequence, the trackers' ability for tracking small objects can be evaluated efficiently. From the video shots in Fig. 10, ours-D can track the *teacan* very accurately. Although the bounding box of ours-D is a little larger than the ground truth, taking the 94th frame as an

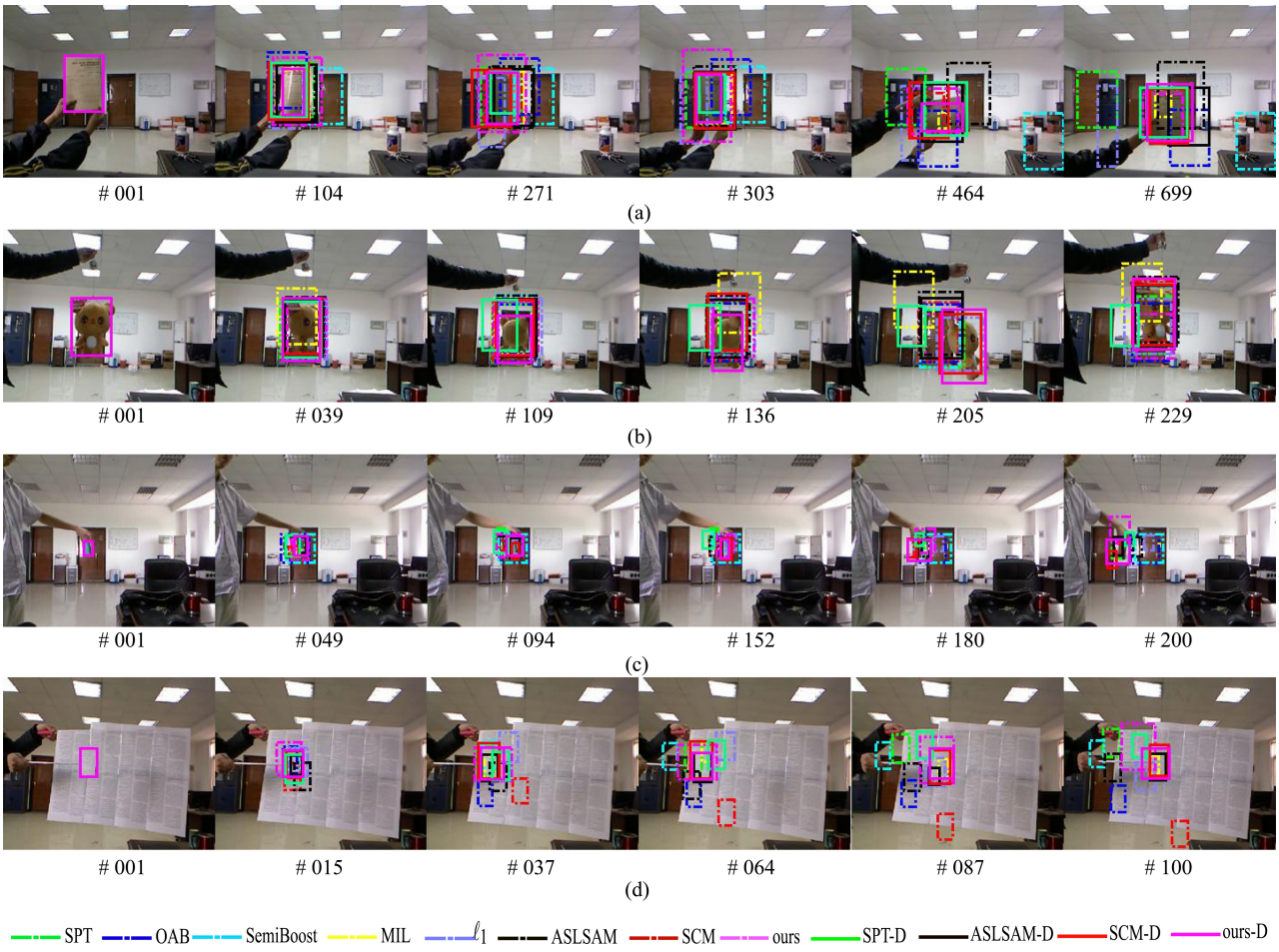


Fig. 10. Some representative experimental results, which are generated by SPT [12], OAB [13], SemiBoost [14], MIL [6], ℓ_1 [8], ASLSAM [15], SCM [9], ours, SPT-D, ASLSAM-D, SCM-D and ours-D, respectively. (a) *Book* sequence. Large variation of shape and illumination. (b) *Bear* sequence. Drastic pose change. (c) *teaCan* sequence. In this sequence, the object is rather small. (d) *Paper* sequence. In this sequence, the background has the similar appearance with the object.

example, the *tea can* can be located in the center of the bounding box for all frames. However, the other trackers all generate drift. For the MIL, SCM, ASLSAM and ℓ_1 trackers, they initialize several positive templates which surround the target center by several pixels away. As for this sequence, the initialized positive templates may contain the part of background clutter owning to the small size of the target. As for the SPT tracker, the segmented superpixels are rather small. The obtained confidence of the superpixels splits into several parts. The problem also exists in our tracker without fusing depth information, but the depth map of the target can guide the proposed tracker to an accurate location. As for the other trackers fusing depth cue, it seems that ASLSAM-D shows a similar tracking performance to ours-D. However, it shows a drift in the 152th frame. Therefore, the proposed tracker excels the other trackers in performance.

4) *Cluttered Background With the Similar Appearance to the Target:* The last sequence *Paper* is designed to evaluate the performance for tackling the cluttered background with similar appearance to the target. In this situation, the trackers modeled in RGB channels have limited ability to distinguish the target with the background. But for the depth map, as shown in Fig. 7, the target' appearance is apparently different from the background. With this superiority, the trackers fusing depth cue are obviously better than the related ones in RGB channels, as well as the proposed tracker. However, from Fig. 9, ours-D absolutely demonstrates better performance than SPT-D, ASLSAM-D and SCM-D. From this phenomenon, the superiority of depth association strategy proposed in this paper is verified. In addition, the scale adaptability of ours-D is apparently better than ASLSAM-D and SCM-D shown in the 87th and 64th frames, respectively.

D. Discussion

In this section, we present two points for further discussion. First, this paper only demonstrates a framework for sparse superpixel-based object tracker via depth fusion. Any other adequate feature representations for superpixel and the depth association model can be embedded in this framework.

Second, the computational cost of the proposed tracker is lower than the existing sparsity-based trackers representing the state-of-the-art, such as ℓ_1, SCM. The detailed analysis is described as follows. The ℓ_1 tracker needs to initialize almost $N = 600$ templates to generate the appearance model. Each template is constructed by pixel-level prototypes. Meanwhile, the number of candidate templates is set as K, ($K \approx 300$).

TABLE III

AverageTimeConsuming (ATC /S) FOR DIFFERENT TRACKERS ON DIFFERENT SEQUENCES. **BOLD** ONES IN THE AVERAGE ROW REPRESENT THE AVERAGE RUNNING TIME OF THE SPARSITY-BASED TRACKERS ON ALL THE SEQUENCES

Trackers	SPT	OAB	SemiBoost	MIL	ℓ_1	ASLSAM	SCM	ours	SPT-D	ASLSAM-D	SCM-D	ours-D
Compiler	Matlab	C++	C++	C++	Matlab	Matlab	Matlab	Matlab	Matlab	Matlab	Matlab	Matlab
book	2.51	0.51	0.71	0.93	3.21	1.52	1.68	0.65	3.42	2.41	2.52	1.42
bear	2.43	0.53	0.77	0.89	3.21	0.75	1.65	0.57	3.62	1.23	2.68	1.56
teaCan	1.32	0.21	0.45	0.64	1.69	0.35	0.98	0.31	2.13	0.89	1.23	0.89
paper	1.45	0.21	0.46	0.84	2.52	0.45	0.59	0.30	2.51	0.51	1.03	1.03
Average	2.41	0.48	0.72	0.95	**3.17**	**1.19**	**1.61**	**0.60**	3.45	**1.89**	**2.42**	**1.45**

The cost is represented as $\mathcal{O}(NM) \times \mathcal{O}(KM)$, where M is the number of the pixels in the template. The SCM also needs many templates to generate its appearance model. In their work, the number of templates is chosen as $n = 250$, which contains 200 negative ones and 50 positive ones. Every template is divided into m ($m << M$) patches. In addition, the number of candidate templates is set as k, ($k \approx 300$). Therefore, the computational cost of SCM is $\mathcal{O}(nm) \times \mathcal{O}(km)$. As for this paper, the sparse superpixel-based appearance model is constructed by U superpixels, where U is the number of superpixels belonging to the target and within [10–100]. The candidate superpixel number V for the new target searching region is within [150–300]. Therefore, the computational cost is $\mathcal{O}(U) \times \mathcal{O}(V)$. From the analysis, it can be seen that the proposed tracker is more computational efficiency.

In order to further analyze the computational cost of different trackers, their actual running time for different sequences is compared in Table. III. It is noted that the number in *Average* row in this table represents the average running time of all the sequences. For a fair comparison, the compiler of each tracker is also listed in the table. From the table, it can be seen that our tracker without exploiting depth cue is the fastest one in the sparsity-based trackers. Meanwhile, after fusing depth cue, the computational cost of the trackers is a little larger than their original ones without fusing depth information. However, the proposed tracker is still the fastest one compared with the other three fusing depth cue.

VII. CONCLUSION

In this paper, a robust superpixel-based tracker via depth fusion is proposed. With the more promising superpixel-based image cues, the target's appearance representation is efficiently constructed. To realize the superpixel-based target estimation, GraphSC is introduced for the first time to constrain the SSDAM, which embeds the geometry relationship of data into the optimal objective function and achieves more powerful discriminative ability. In addition, the depth clue of the target, which provides more discriminative visual information under the cluttered scene, is fused into SSDAM. To prove the tracker's robustness, four challenging video sequences captured by ourselves are contributed to evaluate the trackers' performance under different situations, such as large variation of shape, pose, and illumination, small object and cluttered background with similar appearance to the target. Several competitive trackers representing the state-of-the-art are compared with our tracker, and the comparison results indicate that the proposed tracker is more robust.

In the future, we plan to introduce more clues for the target representation, such as the infrared information. In addition, we also intend to focus on trajectory-based abnormal event detection.

REFERENCES

[1] J. Zhu, Y. Lao, and Y. Zheng, "Object tracking in structured environments for video surveillance applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 2, pp. 223–235, Feb. 2010.

[2] X. Cao, J. Lan, P. Yan, and X. Li, "Vehicle detection and tracking in airborne videos by multimotion layer analysis," *Mach. Vision Appl.*, vol. 23, no. 5, pp. 921–935, Sep. 2012.

[3] J. Guo, Y. Liu, C. Chang, and H. Nguyen, "Improved hand tracking system," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 5, pp. 693–701, May 2012.

[4] J. Gall, A. Yao, N. Razavi, L. Gool, and V. Lempitsky, "Hough forests for object detection, tracking, andaction recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2188–2202, Nov. 2011.

[5] B. Jin, W. Hu, and H. Wang, "Human interaction recognition based on transformation of spatial semantics," *IEEE Signal Process. Lett.*, vol. 19, no. 3, pp. 139–142, Mar. 2012.

[6] B. Babenko, M. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1619–1632, Aug. 2011.

[7] C. Shen, J. Kim, and H. Wang, "Generalized kernel-based visual tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 1, pp. 119–130, Jan. 2010.

[8] X. Mei and H. Ling, "Robust visual tracking using ℓ_1 minimization," in *Proc. IEEE Conf. Comput. Vision*, Sep. 2009, pp. 1436–1443.

[9] W. Zhong, H. Lu, and M. Yang, "Robust object tracking via sparsity-based collaborative model," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2012, pp. 1838–1845.

[10] J. Kwon and K. Lee, "Tracking of a nonrigid object via patch-based dynamic appearance modeling and adaptive basin hopping Monte Carlo sampling," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2009.

[11] S. Nejhum, J. Ho, and M. Yang, "Online visual tracking with histograms and articulating blocks," *Comp. Vis. Image Underst.*, vol. 114, no. 8, pp. 901–914, 2010.

[12] S. Wang, H. Lu, and M. Yang, "Superpixel tracking," in *Proc. IEEE Conf. Comput. Vision*, Jun. 2011, pp. 1323–1330.

[13] H. Grabner and H. Bischof, "On-line boosting and vision," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2006, pp. 260–267.

[14] H. Grabner, J. Matas, L. V. Gool, and P. Cattin, "Tracking the invisible: Learning where the object might be," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2010, pp. 1285–1292.

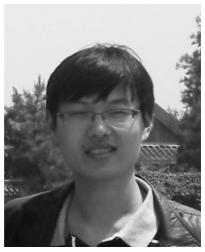
[15] X. Jia, H. Lu, and M. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2012, pp. 1822–1829.

[16] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surveys*, vol. 38, no. 4, pp. 1–45, 2006.

[17] Q. Wang, Y. Yuan, and P. Yan, "Visual saliency by selective contrast," *IEEE Trans. Circuits Syst. Video Technol*., vol. 23, no, 7, pp. 1150–1155, 2013.

[18] F. Lusson, K. Bailey, M. Leeney, and K. Curran, "A novel approach to digital watermarking, exploiting colour spaces," *Signal Process.*, vol. 93, no. 5, pp. 1268–1294, 2013.
[19] S. Avidan, "Ensemble tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, vol. 29, no.2, pp. 261–271, 2007.
[20] J. Kwon and K. Lee, "Visual tracking decomposition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2010, pp. 1269–1276.
[21] Q. Chen, Q. Sun, P. Heng, and D. Xia, "Two-stage object tracking method based on kernel and active contour," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 4, pp. 605–609, Apr. 2010.
[22] J. Yin, C. Fu, and J. Hu, "Using incremental subspace and contour template for object tracking," *J. Network Comput. Appl.*, vol. 35, no. 6, pp. 1740–1748, Nov. 2012.
[23] X. Ren and J. Malik, "Tracking as repeated figure/ground segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2007, pp. 1–8.
[24] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, 2011.
[25] Q. Zhang and K. Ngan, "Segmentation and tracking multiple objects under occlusion from multiview video," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3308–3313, Nov. 2011.
[26] G. García, D. Klein, J. Stückler, S. Frintrop, and A. Cremers, "Adaptive multicue 3-D tracking of arbitrary objects," in *Proc. Pattern Recognit.–Joint 34th DAGM and 36th OAGM Symp.*, pp. 357–366, Aug. 2012.
[27] P. Felzenszwalb and D. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vision*, vol. 59, no. 2, pp. 167–181, 2004.
[28] S. Zhang, H. Yao, X. Sun, and X. Lu, "Sparse coding based visual tracking: Review and experimental comparison," *Patten Recognition*, vol. 46, no. 7, pp. 1772–1788, 2013.
Pattern Recognit., 2012.
[29] M. Zheng, J. Bu, and C. Chun, "Graph regularized sparse coding for image representation," *IEEE Trans. Image. Process.*, vol. 6, no. 1, pp. 1–23, Jan. 2007.
[30] X. Lu, Y. Yuan, and P. Yan, "Robust visual tracking with discriminative sparse learning," *Pattern Recognit.*, vol. 46, no. 7, pp. 1762–1771, 2013.
[31] A. Adam and E. Rivlin, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2006, pp. 798–805.
[32] P. Michel, J. E. Chestnutt, S. Kagami, K. Nishiwaki, J. J. Kuffner, and T. Kanade, "GPU-accelerated real-time 3-D tracking for humanoid locomoiton and stair climbing," in *Proc. IEEE Conf. Intell. Robots Syst.*, Oct. 2007, pp. 463–469.
[33] J. Li, E. Li, Y. Chen, L. Xu, and Y. Zhang, "Bundled depth-map merging for multiview stereo," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2010, pp. 2769–2776.
[34] T. Pitkäaho and T. Naughton, "Calculating depth maps from digital holograms using stereo disparity," *Opt. Lett.*, vol. 36, no. 11, pp. 2035–2037, 2011.
[35] Y. Cong and Y. Tang, "Hand gesture recognition using RGB-D cues," in *Proc. Conf. Inform. Autom.*, Jun. 2012, pp. 311–316.
[36] V. Frati and D. Prattichizzo, "Using kinect for hand tracking and rendering in wearable haptics," in *Proc. IEEE WHC*, Jun. 2011, pp. 317–321.
[37] J. K. M. Park, M. Hasan, and O. Chae, "Hand detection and tracking using depth and color," in *Proc. IEEE Conf. Image Process. Comput. Vision Pattern Recognit.*, Jul. 2012, pp. 779–785.
[38] G. Fanelli, T. Weise, J. Gall, and L. Gool, "Real time head pose estimation from consumer depth cameras," in *Proc. Annu. Symp. German Assoc. Pattern Recognit.*, Sep. 2011, pp. 101–110.
[39] S. A. Teichman, "Tracking-based semi-supervised learning," in *Int. J. Robot. Res.*, vol. 37, no. 7, pp. 804–818, 2012.
[40] R. Achanta, A. Shaji, K. Smith, A. Lucchi, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2281, Nov. 2012.
[41] S. Todorovic and N. Ahuja, "Region-based hierarchical image matching," *Int. J. Comput. Vision*, vol. 78, no. 1, pp. 47–66, 2008.
[42] E. Erdem, S. Dubuisson, and I. Bloch, "Fragments based tracking with adaptive cue integration," *Comp. Vis. Image Underst.*, vol. 116, no. 7, pp. 827–841, Jul. 2012.
[43] H. Lee, A. Battle, and A. Y. Ng, "Efficient sparse coding algorithms," in *Proc. Advances Neural Inform. Process. Syst.*, pp. 801–808, Dec. 2007.
[44] X. Li, A. Dick, H. Wang, C. Shen, and A. Hengel, "Graph mode-based contextual kernels for robust SVM tracking," in *Proc. IEEE Conf. Comput. Vision*, Nov. 2011, pp. 1156–1163.
[45] T. Chan and W. Zhu, "Level set based shape prior segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2005, pp. 1164–1170.

Yuan Yuan (M'05–SM'09) is a Full Professor with the Center for OPTical IMagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, Shaanxi, China. She has published over 100 papers, including in reputable journals, such as the IEEE TRANSACTIONS AND PATTERN RECOGNITION, as well as conferences papers in CVPR, BMVC, ICIP, ICASSP, etc. Her research interests include visual information processing and image/video content analysis.



Jianwu Fang received the B.E. degree in automation and the M.E. degree in traffic information engineering and control from Chang'an University, Xi'an, China, in 2009 and 2012, respectively. He is currently pursuing the Ph.D. degree at the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, Shaanxi, China.

His research interests include computer vision and pattern recognition.

Qi Wang received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent system from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively.

He is currently an Associate Professor with the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, Shaanxi, China. His research interests include computer vision and pattern recognition.