# Dictionary Learning and Confidence Map Estimation-Based Tracker for Robot-Assisted Therapy System

Xiaolong Zhou[1,2](✉), Sixian Chan[2], Junwei Li[3], Shengyong Chen[2,4], and Honghai Liu[5]

[1] Quzhou University, Quzhou 324000, China
xlvision@hotmail.com
[2] Zhejiang University of Technology, Hangzhou 310023, China
[3] Huawei Technologies Co., Ltd., Hangzhou 310023, China
[4] Tianjin University of Technology, Tianjin 300384, China
[5] University of Portsmouth, Portsmouth PO1 2UP, UK

**Abstract.** In this paper, we propose a new tracker based on dictionary learning and confidence map estimation for a robot-assisted therapy system. We first over-segment the image into superpixel patches, and then employ color and depth cues to estimate the object confidence of each superpixel patch. We build two Bag-of-Word (BoW) models from initial frames to encode foreground/background appearance, and compute object confidence at superpixel level using BoW model in both foreground and background. We further refine target confidence by depth-based statistical features to mitigate noise interference and the uncertainty of visual cues. We derive the global confidence of each target candidate at bag level, and incorporate the confidence estimations to determine the posterior probability of each candidate within the Bayesian framework. Experimental results demonstrate the superior performance of the proposed method, especially in long-term tracking and occlusion handling.

**Keywords:** Object tracking · RGB-D · Bag-of-Word · Occlusion handling

## 1 Introduction

Tracking and analyzing the behavior of the patients, in particular to track the objects in interaction plays a significant role in a Robot-Assisted Therapy (RAT) system. Although many excellent tracking methods have been proposed, issues of robustness and reliability make these methods unsuitable for real-world situations. One main cause of failing tracking is the degradation of the tracking model, where the accumulation of inaccurate tracking results and corresponding model update over a period of time may cause focus to drift from the subject.

Object tracking has witnessed great advance in both generative and discriminative branches [1]. One of the advantages of generative model is that the method models target appearance without large number of training samples, however, it cannot separate target effectively from clutter background, occlusion, and long-term period. Discriminative tracking methods [2–5] achieve superior performance both in success

© Springer Nature Switzerland AG 2019
Z. Lin et al. (Eds.): PRCV 2019, LNCS 11857, pp. 147–159, 2019.
https://doi.org/10.1007/978-3-030-31654-9_13

and precision rate thanks to the combination of robust feature representation, discriminative classifier and the exploit of background and foreground information. However, these methods cannot well handle clutter background and sudden target appearance variation problems. Recently, trackers proposed by combining Convolutional Neural Network (CNN) and Discriminative Correlation Filter (DCF) achieve state-of-the-art performance due to the representation of deep feature and the computational efficiency of DCF. For instance, one or multiple layer deep features have been employed to train the DCF in frequency domain and thus can get good performance [6]. Tao et al. [7] employ two Siamese CNN to learn a generic matching function for tracking task to handle appearance variation and lacking of training samples. David et al. [8] propose to learn a regression function by CNN from large annotated video sequence, which achieves a very high tracking speed (more than 100FPS). In recent years, superpixel-based tracking methods [9–11] have attracted extensive attentions due to high accuracy and robustness. Yang et al. [9] compute a target-background confidence map using discriminative appearance model based on superpixels, and obtain the best candidate by maximizing a posterior estimate. In [10], a Dynamic Graph-based Tracker (DGT) is built to model the superpixel interactions, the tracking problem is then posed as a matching problem between the target graph and the candidate graph. In [11], tracking method based on Bag-of-Word (BoW) model is proposed to estimate target confidence, however, the method cannot handle the image patches with similar appearance to both background and foreground, which make tracker prone to degradation over-time and not appropriate for long-term tracking.

The aforementioned trackers either suffer from lacking background information supervision, computational burden, or cannot handle noisy patch interference, which are inappropriate for accurate real-world tracking. To meet the goal of long-term robust target tracking for the RAT system, this paper proposes a discriminative model-based tracking method to achieve robust and accurate tracking even for long-term period by fusing color and depth information to estimate target confidence.

## 2 Proposed Tracking Method

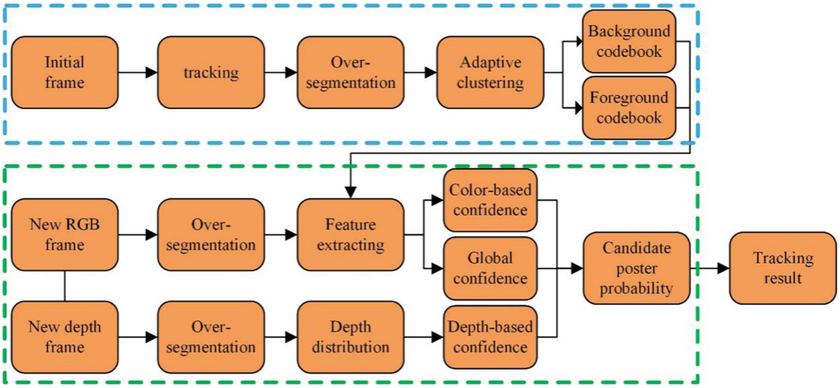

**Fig. 1.** Pipeline of the proposed tracking method.

The architecture of the proposed tracking method is illustrated in Fig. 1. The target object and background patches are over-segmented into superpixel collections, and then an adaptive AP is employed to select discriminative superpixel patches as background/foreground codebook. When a new frame arrives, we derive the posterior probability of each target candidate from three aspects within the Bayesian framework. To overcome model degradation over time, we propose a new update strategy based on the inspiration that the best appearance model could minimize the reconstruction error of current target.

## 2.1 Superpixel Appearance and BoW Model Construction

To compute the confidence of each superpixel, we construct foreground and background discriminative appearance model based on superpixel segmentation and BoW theory. Given the training image set $\{I_1, I_2, \ldots\ldots, I_k\}$, where $k$ is number of training image, we first over-segment them into a set of superpixel patches using SLIC algorithm [12], and then the normalized color histogram $f_i^k$ in the HSI space is extracted as an appearance representation descriptor, where $i$ is superpixel index in $I_k$. HIS color histogram is employed because of its robustness in handling light changes and its discriminative ability in feature representation. Those superpixels inside the target area $O_k$ are treated as positive training patches, while those outside the area of $O_k$ but within 2 × $O_k$ are treated as negative training patches. The annular band can be expressed as $S_k$.

Given the feature set $\left\{f_i^k\right\}$ of target superpixel collection, the codebook of BoW model is generated by performing clustering, and cluster centers are used to initialize the codebook. In [11], the authors employ $k$-means algorithm to perform feature center selection. However, it needs to specify seed points manually. To remedy this, we utilize the Affinity Propagation (AP) clustering method [13] to determine the feature centers, which can facilitate two advantages: (1) it has the ability to determine the number of cluster centers automatically and (2) it is computation efficient. The input of AP is the affinity matrix of $\mathrm{S} \subseteq R^{N \times N}$. Each data point of $S(i,j) = -\left\|f_i^K - f_j^k\right\|^2$ is defined as the negative Euclidean distance between $f_i^k$ and $f_i^k$. By viewing each data point as a node in a network, the cluster centers and the corresponding exemplars are emerged by recursively transmit real-value distances along edges.

After the AP clustering, we can get two sets of codebook $F_m = \left\{F_1^F, F_2^F, \ldots, F_m^F\right\}$ and $B_n = \left\{F_1^B, F_2^B, \ldots, F_n^B\right\}$ corresponding to feature centers of background and foreground superpixel training sets, in which $m$ and $n$ denote the length of codebook, and the superscripts $F$ and $B$ correspond to foreground and background sets. The superpixels in $O_k$ and $S_k$ of each training samples are assigned to the nearest elements in $F_m$ and $B_n$ respectively, by minimizing Eqs. (1) and (2)

$$L_n^F = \arg\min_i \| f_n^k - F_i^F \|, f_n^k \in O_k \tag{1}$$

$$L_m^B = \arg\min_i \| f_m^k - F_i^B \|, f_m^k \in S_k \tag{2}$$

where $f_n^k$ is the $n$-th superpixel feature vector of the $k$-th training image. $L_n^F$ and $L_m^B$ denote index of the $n$-th superpixel assigned to the word in codebook. Two histograms $H^F(I)$ and $H^B(I)$ are generated corresponding to the foreground and background bags, which indicate the occurrence frequency of each codeword in $k$ training images. $F_i^F$ and $F_i^B$ denote the $i$-th codewords of foreground and background codebook, respectively.

### 2.2 Local Background-Foreground Confidence Estimation

One challenge to estimate the foreground and background confidence of each superpixel is the interference of those superpixels which are inside the foreground rectangle patch but not belong to the target, which we name them as false-positive superpixels. So, the first step of our confidence estimation is to remove the impact of false-positive superpixels. For a test image $I$, we segment it into a set of superpixels $\mathrm{SP} = \{\mathrm{sp}(1), \mathrm{sp}(2), \ldots, \mathrm{sp(k)}\}$, and then compute two distances $\{d^F(\mathrm{i}), d^B(\mathrm{i})\}$ between the $i$-th superpixel and the nearest codeword in $F_n^F$ and $F_m^B$. Let $d^F(i,m)$ and $d^B(i,m)$ be the superpixel similarity to foreground and background codewords.

$$d^F(i,m) = \exp(- \parallel sp^k(i) - F_m^F \parallel_2^2) \tag{3}$$

$$d^B(i,n) = \exp(- \parallel sp^k(i) - F_n^B \parallel_2^2) \tag{4}$$

The similarity of $sp(k)$ to the nearest codeword is obtained by minimizing

$$d^F(i) = \min_m d^F(i,m), \text{and}\, d^B(i) = \min_n d^B(i,n) \tag{5}$$

We define a false-positive superpixel based on rules in Eq. (6)

$$M(i) = \begin{cases} -1 & if\ d^F(i) \le d^B(i) \\ 0\ if\ d^F(i) \ge & th^F, d^B(i) \ge th^B \\ 1 & if\ d^F(i) \ge d^B(i) \end{cases} \tag{6}$$

where $th^F$ and $th^B$ represent the outlier thresholds of foreground and background. If $M(i) = 0$, then the $i$-th superpixel is an ambiguity one, otherwise, it belongs to either foreground ($M(i) = 1$) or background ($M(i) = -1$). We assign foreground and background confidence of each superpixel based on the combination of bag similarity and superpixel distance. In Eq. (5), each superpixel in SP is assigned to the nearest codeword in codebook $F_m$ and $B_n$. So, it is easy to compute bag histogram distribution of $B^F(I)$ and $B^B(I)$. Therefore, two bag similarities (see Eqs. (7) and (8)) can be determined.

$$S^F = \min_{l \in [1,k]} \{\exp(- \parallel B^F(I) - H_l^F \parallel)\} \tag{7}$$

$$S^B = \min_{l \in [1,k]} \{\exp(- \parallel B^B(I) - H_l^B \parallel)\} \tag{8}$$

where $H_l^F$ and $H_l^B$ denote background and foreground BoW histograms of the $l$-th positive and negative training sample. The two similarities indicate the target background confidence of a sample at bag level.

To further refine the confidence of each superpixel patch, a local confidence value $C(i) \in (0; 1)$ is assigned based on $S^F$ and $S^B$. The value is computed as follows.

$$C(i) = M(i) * w(I) * \max\{d^F(i), d^B(i)\} \tag{9}$$

$$w(I) = \frac{S^F}{S^F + S^B} \tag{10}$$

where $w(I)$ denotes the weighting term of the sample image $I$ belonging to target. The local confidence of $C(i)$ is determined jointly by $M(i)$, $w(I)$ and $max\{d^F(i), d^B(i)\}$. $M(i)$ is used to distinguish which category of the $i$-th superpixel belongs to. $w(I)$ corresponds to the weighting term of image $I$ belonging to target, which is defined in Eq. (10). $max\{d^F(i), d^B(i)\}$ indicates the likelihood of the superpixel. It should be noted that the confidence of ambiguity superpixel patches is set to zero, which means that the local feature based on superpixel patch is not enough to estimate target-background confidence.

## 2.3 Depth-Based Confidence Estimation

For a superpixel patch which is difficult to estimate target confidence from appearance model, by incorporating the depth feature we can predict its category easily. However, only relying on the depth cue is still not enough to predict which category of a superpixel belongs to due to the fact that the depth is weak in encoding target texture feature. To remedy this, we employ both depth cue and appearance model to estimate the confidence of false-positive superpixel and refine the confidence of the other superpixels.

Instead of estimating superpixel confidence directly, we propose to use the aforementioned AP clustering result to compute the confidence of each cluster. Then the cluster center $F_m^F$ and its member set $\{sp_m(k)\}$, where $k$ is the superpixel index, correspond to their own image regions in training samples. Here, we compute two scores $\mathrm{R}_{in}(i)$ and $\mathrm{R}_{out}(i)$ for each cluster and its corresponding members. $\mathrm{R}_{in}(i)$ denotes the area of the $i$-th cluster and its members overlapping the target area. $\mathrm{R}_{out}(i)$ indicates the superpixel area out of the target region. The cluster confidence is defined as Eq. (11).

$$C_{clust}(i) = \frac{R_{in}(i) - R_{out}(i)}{R_{in}(i) + R_{out}(i)} \tag{11}$$

where $\mathrm{C}_{clust}(i) \subseteq [-1, 1]$, higher value indicates that the superpixel clustering owns higher confidence belonging to target, otherwise, the clustering is more likely to belonging to background.

Then, we compute the depth mean and standard deviation of each cluster as the depth model to constraint the background and foreground confidence. Let

$$mean_m(i) = \frac{1}{k}\sum_{k=1}^{K} depth(sp(k)) \tag{12}$$

and

$$std_m(i) = \sqrt{\frac{1}{k}\sum_{k=1}^{K} (depth(k) - mean(i))^2} \tag{13}$$

be the depth mean and standard deviation of cluster $F_M^F$ and the corresponding superpixel set $\{sp_m(k)\}$. Intuitively, for the superpixel patches belonging to the same clustering, their depth distribution should be uniform and the standard deviation is expected to be small. Although the depth feature of superpixel lacks discriminative capacity and semantic information to estimate confidence, it is an important cue to predict target confidence based on the prior knowledge of the homogeneity of the depth distribution and the continuity of depth changing.

The confidence value of each superpixel patch with depth constraint is defined as

$$C_{depth}(i) = w_{depth}(i,m) * C_{clust}(i) \tag{14}$$

$$w_{depth}(i,m) = \exp(-\lambda_d \times \frac{|depth(i) - mean_m(i)|}{std_m(i)}) \tag{15}$$

where $\mathrm{w}_{depth}(i,m)$ is the constraint term and follows the Gaussian distribution. Greater distance to mean cluster depth indicates lower likelihood of the superpixel belonging to the foreground, the pairwise index of $i$ and $m$ means that the $i$-th superpixel is assigned to the $m$-th cluster by Eqs. (1) and (2).

## 2.4 Global Confidence Estimation

The previous appearance-based model and depth-based model are used to determine the confidence of a certain superpixel. Now we use bag similarity to compute global confidence of a test sample. When a target candidate arrives, we first segment it to a set of superpixels $sp(i)$, $i \in \{1, 2, \ldots, \mathrm{N}\}$, where N is the number of superpixels. Then, we assign each superpixel patch to the nearest codeword to compute two candidates' bags (codeword distribution) corresponding to background histogram $\mathrm{H}_t^B(I)$ and foreground histogram $\mathrm{H}_t^F(I)$. Two similarities $S^F(I)$ and $\mathrm{S}^B(I)$ are employed to measure the candidate $I$ belonging to background or foreground, and then they are considered to determine the global candidate confidence $\mathrm{C}_{global}(I)$ jointly.

$$C_{global}(I) = \frac{S^F(I) - S^B(I)}{S^F(I) + S^B(I)} \tag{16}$$

$$S^F(I) = \exp(-\lambda_f \times \parallel H_t^F(I) - H^F(I) \parallel_2^2) \tag{17}$$

$$S^B(I) = \exp(-\lambda_b \times \parallel H_t^B(I) - H^B(I) \parallel_2^2) \tag{18}$$

The global confidence ranges from −1 to 1. When the similarity of candidate image becomes similar to the background model, its confidence value is close to −1, the confidence value is close to 1 if it is similar to the foreground model. Different from other global confidence estimation methods, two BoW models are used to estimate the target confidence, which is robust in dealing with the ambiguous candidates. In other words, when the candidate is close to both foreground and background models, its confidence of being the target is close to 0.

## 2.5 The Proposed Tracking Method

Given the target observation set $\mathrm{Y}^t = \left\{y_1^t, y_2^t, \ldots, y_n^t\right\}$ at frame $t$, where $y_n^t$ denotes the $n$-th observation of target at the $t$-th frame. We perform tracking by maximizing the posteriori probability in Eq. (19).

$$\hat{X}_t = \arg\max_{x_t^i} p(X_t^i | Y^t) \tag{19}$$

where $\mathrm{X}_t^i$ stands for the $i$-th target candidate state of frame $t$, and $\mathrm{Y}^t$ denotes the corresponding observation of $\mathrm{X}_t^i$. In this paper, we define the target state as $X_t = \left\{X_t^c, X_t^{sx}, X_t^{sy}\right\}$, where $X_t^c$, $X_t^{sx}$, and $X_t^{sy}$ represent the target center location, target scales in $x$-axis and $y$-axis, respectively. The posterior probability of the given observation set $Y^t$ up to frame $t$ is achieved by the Bayesian theorem recursively.

$$p(X_t | Y_t) \propto p(Y_t | X_t) \int p(X_t | X_{t-1}) p(X_{t-1} | Y_{t-1}) dX_{t-1} \tag{20}$$

where $p(Y_t|X_t)$ and $p(X_t|X_{t-1})$ denote the observation model and motion model respectively. The motion model indicates the relationship between target state and frames in time domain, and we assume that it follows the Gaussian distribution. Thus, the target state variation can be formulated as Eq. (21).

$$p(X_{t-1} | Y_{t-1}) = N(X_t; X_{t-1}, \Psi) \tag{21}$$

where $\Psi$ is a diagonal covariance matrix, and the elements in $\Psi$ denote the standard deviation of target state. The observation model is formulated based on the sum of appearance confidence in Eq. (9), depth confidence in Eq. (14) and the corresponding global target confidence at bag level in Eq. (16). When the target location of frame $t - 1$ has been determined, we select a rectangle $R_t$ area around the previous target center as the searching space in the $t$-th test image. To reduce computation load, we only over-segment image into superpixels within $R_t$ once. For each candidate target state in $\mathrm{X}_t^i$, we assign the corresponding superpixel set to it, and then approximate the confidence based on the assigned superpixel collections.

$$p(Y_t|X_t) \propto C_{global}(I) + \sum_{i \in \Omega} (C(i) + C_{depth}(i)) \tag{22}$$

where $\Omega$ denotes the superpixel set when the target state is set to $\mathrm{X}_t$. The state observation estimation is proportional to confidence sum in Eq. (22).

It is essential to update model effectively for capturing target appearance variation due to pose change, illumination change, and occlusion *et al*. In this paper, the words in codebook play an important role in encoding target appearance. So, the way to select and update discriminative words in both temporal and spatial domain is particular important. We assume that the best update strategy is to select words that can minimize the reconstruction error of the current target. Based on this inspiration, we propose a simple and effective sparse representation method to select the most discriminative words from the previous frames to estimate target state of current frame.

In order to effectively use the depth distribution to reduce the uncertainty of superpixel appearance, we update the mean and standard deviation of each cluster based on the depth distribution of the tracked target every frame. We use a temporal low-pass filtering method to accommodate target depth distribution variation.

$$mean_m^*(i) = (1 - \rho_1) mean_m(i) + mean_k(i) \tag{23}$$

$$std_m^*(i) = (1 - \rho_2) \mathrm{std}_m(i) + std_k(i) \tag{24}$$

where $\mathrm{mean}_k(i)$ and $\mathrm{std}_k(i)$ denote the mean depth and standard deviation of the $k$-th frame target area.

## 3 Experimental Results and Analysis

Six challenging video sequences with RBG color channel and depth channel are captured by our RAT system, namely *Bear, Bear2, Wolf, Wolf2, Ballon, Dog*. Both the RGB channel and depth channel are recorded by a Kinect sensor and calibrated to the same coordinate system. We annotate the target bounding box manually by a rectangle in each image, and then the rectangle is projected to depth image as annotation. The annotation in RGB and depth channels is treated as groundtruth to evaluate the performance of our tracker. Each of the recorded video sequence contains at least one challenge such as occlusion, shape deformation, rotation, etc. The length of each video is 800, 952, 731, 1027, 1109 and 1210 frames, respectively.

We use SLIC algorithm to over-segment image in HSI color space and employ a KCF with HOG feature to track the initial four image frames (from the second to the fifth). A total of 5 frames are used to construct the BoW model. When performing AP clustering, we employ negative Euclidean distance as the real value information. The exemplar preference is set to 1.5 times of the average negative Euclidean distance. We update the codewords every 5 frames and update the depth model every frame.

### 3.1 Codewords Extracted by AP

To verify the impact of the number of codewords on appearance model, we implement varieties of $F_m$ and $B_m$ by setting different codeword numbers. The target and background appearances are encoded by codewords in $F_m$ and $B_m$. We employ the AP method to select representative superpixel patches as feature centers adaptively, which overcomes the deficiency of generating seed points manually. However, another parameter, the number of codewords in $F_m$ and $B_m$, is considered highly important. In AP cluster method, the number of clusters is influenced by a real value *s(k,k)*, which is referred as "preference" for each feature vector *k*. So, the feature vector with larger values of *s(k,k)* is more likely to be chosen as a cluster center. As a priori knowledge, all the feature vectors are equally considered as cluster center candidates, and we set a common "preference" to each superpixel patch as initial state. The shared value can be varied to produce different numbers of clusters. Specifically, we will get a moderate number of clusters when set *s(k,k)* to the median of input similarities. If the shared value is set to a smaller value than the median of input similarities, it would result in a smaller number of clusters.

**Table 1.** Analysis of AP preference on target appearance model in video sequence wolf

| s(k,k) | Superpixel patch | AP clusters | AUC of success | AUC of precision |
|---|---|---|---|---|
| 1.0 | 382 | 64 | 0.68 | 0.89 |
| 0.8 | 382 | 52 | 0.642 | 0.875 |
| 0.6 | 382 | 35 | 0.61 | 0.85 |
| 0.4 | 382 | 22 | 0.59 | 0.835 |
| 1.2 | 382 | 71 | 0.67 | 0.865 |
| 1.4 | 382 | 80 | 0.665 | 0.86 |
| 1.6 | 382 | 101 | 0.64 | 0.85 |

In Table 1, we build different target appearance codebooks by setting *s(k,k* = *{1.0,0.8,0.6,0.4,1.2,1.4,1.6}* in Wolf video sequence. *s(k,k)* = 1.0 denotes that we set the preference value to 1.0 times the median similarity. It can be seen that the AP cluster number increases with the value of *s(k,k)*. As *s(k,k)* increases from 1 to 1.6, the number of clustering centers increases from 64 to 101, however, the AUC of success plot and precision plot are reduced by 5.8% and 4.4%, respectively. On the contrary, the cluster center number drops to 35 from 64 with respect to *s(k,k)* varying from 1.0 to 0.4, while the corresponding tracking performance is reduced by about 13.2% (Success) and 6.2% (Precision). This indicates that the number of codewords plays an important role in appearance model. Too many codewords undermine the discriminative ability of the appearance model, while insufficient codewords is not robust to target appearance variation.

### 3.2 Effectiveness of the Background/Foreground Appearance Model

Contrary to existing target confidence map estimation methods, we propose to use background and foreground appearance model to perform confidence estimation jointly. We design the dual models mainly considering the disturbance of ambiguous superpixel patches. As a priori, when a superpixel is similar to both background and foreground, we consider it will undermine the model representative ability. Moreover, the confidence map based on these ambiguous superpixels is unreliable. Inspired by this observation, we propose to build a robust background-aware target appearance model.

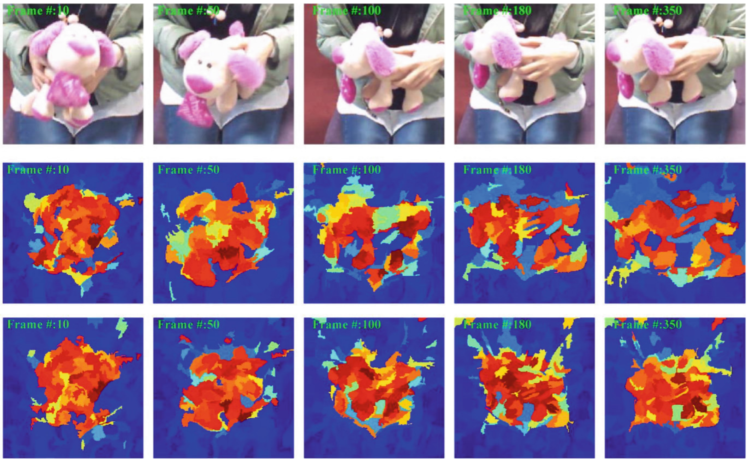

**Fig. 2.** Confidence estimation results between dual appearance model and target-based appearance model. The first row indicates the RGB image captured from the Kinect sensor. The second and the third rows indicate the corresponding confidence maps of a single appearance model (the second row) and a dual appearance model (the third row).

As shown in Fig. 2, the target appearance model based on background and foreground achieves excellent precision in predicting target confidence. From frame 10 to 350, the target experiences significant appearance variation (rotation and occlusion). The proposed dual model can identify ambiguous and noisy superpixel patches and then prevent them to participate in appearance mode building. On the other hand, the dual model selects the most discriminative superpixel automatically to encode target appearance, which is an effective method to prevent model degradation as well as keeps model robust to distractors. On the contrary, the confidence map in the second row of Fig. 2 is the result estimated from only target appearance model. In other words, we complete another appearance model with the same method to dual model, the main difference is that only target appearance (without considering background context) is used to compute the confidence map. The confidence map of the 100th frame has shown a significant deviation, with the increase in the number of frames, this error is gradually accumulated and results in model degradation. At frames 180 and 350, the confidence map from dual model still shows a high precision compared to the groundtruth in the first row. However, the confidence map corresponding to frames 180 and 350 in the second row starts to show large deviation and discontinuity, while the confidence margin between the background part and target becomes smaller.

### 3.3 Quantitative Analysis

We use two protocols to evaluate the tracking performance: area under curve (AUC) of one-pass evaluation (OPE) using success plot, and center location error (CLE). The success plot is used to measure the overlap rate between tracked bounding box and the grountruth on a sequence of video frames. The later metric denotes the distance between the tracked target center and groundtruth center.

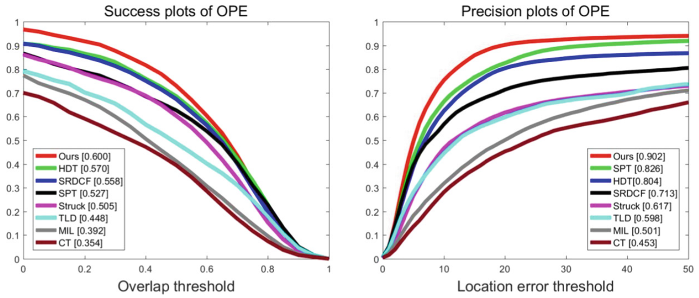

**Fig. 3.** Average success plot (left) and precision plot (right) of OPE on our own video sequences datasets.

We evaluate our tracker against 7 state-of-the-art trackers including SPT [9], HDT [14], CT [15], MIL [16], TLD [4], Struck [17], and SRDCF [18]. Among them, the SPT tracker is a superpixel-based tracking method, the SRDCF is a state-of-the-art KCF-based tracking method, the HDT is a hierarchical convolutional neural network and correlation filter based method, and the other trackers are selected due to their excellent performance in OTB benchmark. The quantitative evaluation results between the proposed tracker and the state-of-the-art trackers are show in Fig. 3. The success plot shows that our tracker outperforms all of the other trackers with a large margin. Comparing with the second-best tracker SRDCF and the third best tracker HDT, the proposed tracking method obtains the AUC of 0.652 and improves the performance of SRDCF and HDT by about 12% and 16%, respectively. Both the proposed tracker and SPT tracker make use of superpixel confidence to track object, however, our method shows an obvious improvement in terms of the both success and precision. As to center location error, our method is still superior to the rest trackers. Table 2 summarizes the average CLE of each tracker. The results demonstrate that the proposed tracker achieves the best performance with the minimum average CLE on all video sequences.

**Table 2.** Quantitative comparison of average CLE

| Sequence | Ours | SPT | HDT | CT | MIL | TLD | Struck | SRDCF |
|---|---|---|---|---|---|---|---|---|
| Bear | 4.2 | 5.9 | 7.8 | 7.9 | 8.6 | 6.9 | 7.2 | 7.2 |
| Bear2 | 3.8 | 4.1 | 4.4 | 5.0 | 4.2 | 7.2 | 5.8 | 25.9 |
| Wolf | 4.6 | 9.7 | 6.9 | 8.0 | 7.4 | 60.8 | 7.6 | 11.2 |
| Wolf2 | 3.7 | 7.7 | 5.4 | 5.9 | 6.1 | 4.3 | 9.5 | 4.8 |
| Ballon | 6.3 | 9.5 | 8.5 | 6.9 | 9.2 | 11.1 | 11.3 | 9.0 |
| Dog | 3.5 | 7.6 | 6.0 | 11.1 | 9.1 | 8.9 | 8.4 | 8.8 |

## 4 Conclusion

In this paper, we have presented a novel object tracking method using RGB and depth images from the Kinect sensor for a RAT system. We trained two BoW models to encode the target background appearance, and combined depth distribution to refine tracking result. To achieve accurate and long-term tracking, we computed two target confidence maps based on color and depth information at superpixel level, and computed a global confidence of each target candidate using codewords. Furthermore, our tracking method was equipped with a sparse representation-based discriminative online update strategy to handle with target appearance variation and occlusion. Experiments on six video sequences have showed that the proposed tracking method outperformed the state-of-the-art tracking methods in both success and precision plots. Moreover, our method can prevent tracking model degradation effectively which is suitable for long-term tracking and real-world application.

**Acknowledgement.** This work was supported by National Key R&D Program of China (2018YFB1305200), National Natural Science Foundation of China (61876168, U1509207, 61733011, 51575338), and Zhejiang Provincial Natural Science Foundation of China (LY18F030020).

## References

1. Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A., Hengel, A.V.D.: A survey of appearance models in visual object tracking. ACM Trans. Intell. Syst. Technol. **4**(4), 1–48 (2013)
2. Zhou, X., Li, J., Chen, S., Cai, H., Liu, H.: Multiple perspective object tracking via context-aware correlation filter. IEEE Access **6**(1), 43262–43273 (2018)
3. Zhou, X., Li, Y., He, B., Bai, T.: GM-PHD-based multi-target visual tracking using entropy distribution and game theory. IEEE Trans. Industr. Inf. **10**(2), 1064–1076 (2014)
4. Kalal, Z., Matas, J., Mikolajczyk, K.: P-n learning: bootstrapping binary classifiers by structural constraints. In: IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, USA, pp. 49–56. IEEE Press (2010)
5. Chan, S., Zhou, X., Li, J., Chen, S.: Adaptive compressive tracking based on locality sensitive histograms. Pattern Recogn. **72**, 517–531 (2017)
6. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 583–596 (2015)

7. Tao, R., Gavves, E., Smeulders, A.W.: Siamese instance search for tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, pp. 1420–1429. IEEE Press (2016)
8. Held, D., Thrun, S., Savarese, S.: Learning to track at 100 FPS with deep regression networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 749–765. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_45
9. Yang, F., Lu, H., Yang, M.-H.: Robust superpixel tracking. IEEE Trans. Image Process. **23**(4), 1639–1651 (2014)
10. Wen, L., Du, D., Lei, Z., Li, S. Z., Yang, M.-H.: Jots: joint online tracking and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, Massachusetts, pp. 2226–2234. IEEE Press (2015)
11. Fan, H., Xiang, J., Zhao, L.: Robust visual tracking via bag of superpixels. Multimedia Tools Appl. **75**(14), 8781–8798 (2016). https://doi.org/10.1007/s11042-015-2790-3
12. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S.: SLIC superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. **34**(11), 2274–2282 (2012)
13. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. Science **315**(5814), 972–976 (2007)
14. Qi, Y., Zhang, S., Qin, L., Yao, H., Huang, Q., Lim, J., Yang, M.-H.: Hedged deep tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, pp. 4303–4311. IEEE Press (2016)
15. Zhang, K., Zhang, L., Yang, M.-H.: Real-time compressive tracking. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7574, pp. 864–877. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33712-3_62
16. Babenko, B., Yang, M.-H., Belongie, S.: Robust object tracking with online multiple instance learning. IEEE Trans. Pattern Anal. Mach. Intell. **33**(8), 1619–1632 (2011)
17. Hare, S., Golodetz, S., Saffari, A., Vineet, V., Cheng, M.-M., Hicks, S.L., Torr, P.H.: Struck: structured output tracking with kernels. IEEE Trans. Pattern Anal. Mach. Intell. **38**(10), 2096–2109 (2016)
18. Danelljan, M., Hager, G., Shahbaz Khan, F., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, pp. 4310–4318. IEEE Press (2015)