Robust Object Tracking in Crowd Dynamic Scenes using Explicit Stereo Depth

Chi Li[1], Le Lu[2], Gregory D. Hager[3], Jianyu Tang[1], Hanzi Wang[1]

[1] Cognitive Science Department, Xiamen University, Xiamen, China
[2] Image Analytics and Informatics, Siemens Corporate Research, Princeton, NJ, USA
[3] Computer Science Department, Johns Hopkins University, Baltimore, MD, USA

Abstract. In this paper, we exploit robust depth information with simple color-shape appearance model on single object tracking in crowd dynamic scenes. Since binocular video streams are captured from a moving camera rig, background subtraction cannot provide a reliable enhancement of region of interest. Our main contribution is a novel tracking strategy to employ explicit stereo depth to track and segment object in crowd dynamic scenes with occlusion handling. Appearance cues including color and shape play a secondary role to further extract the foreground acquired by previous depth-based segmentation. The proposed depth-driven tracking approach can largely alleviate the drifting issue, especially when the object frequently interacts with similar background in long sequence tracking. The problems caused by rapid object appearance change can also be avoided due to the stability of the depth cue. Furthermore, we propose a new, yet simple and effective depth-based scheme to cope with complete occlusion in tracking. From experiments on a large collection of challenging outdoor and indoor sequences, our algorithm demonstrates accurate and reliable tracking performance which outperforms other state-of-the-art competing algorithms.

1 Introduction

Although much progress has been made in recent years, object tracking still remains an unsolved problem since most appearance-based tracking approaches [1–6] still fail to eliminate the drift problem caused by similar background outliers. Moreover, occlusion is hard to be discriminated from sharp appearance changes using appearance cues, which results in the failure of occlusion detection. And there is no effective method to reacquire an object after complete occlusion. Compared with appearance cues, depth information that encodes 3D spatial relationships among objects is more helpful for achieving robust tracking. To make depth cue applicable to most of tracking situations, we use an improved generic stereo algorithm [7] to generate dense depth map from a binocular video sequence. Our tracking scenario is similar to [8–10] which is extremely challenging due to various factors such as motion blur, varying lighting, continuously interacting moving objects, frequent partial and complete occlusion and mobile camera placement. We exploit *discrete object-depth labels* on generated depth

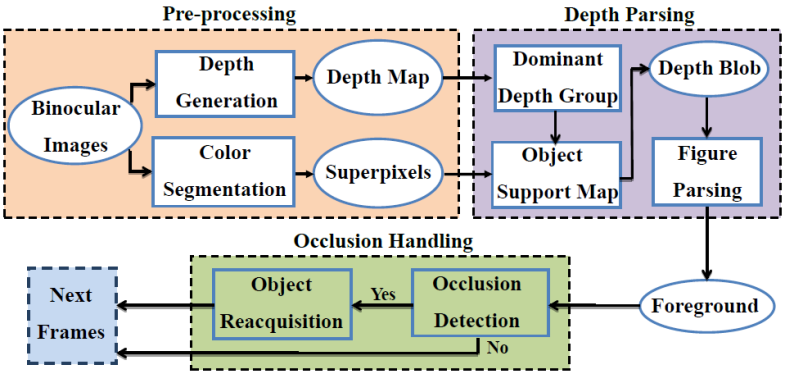

Fig. 1. Depth-Driven Tracking Framework.

map repetitively over time to drive the object tracking and segmentation in crowd dynamic scenes, based on the assumption of *statistically uniform depth distribution within the object area.*

In our depth-driven tracking framework, object tracking is treated as the propagation of the spatial object blob on a uniform depth support map. We observe that depth change is more stable than other appearance cues between consecutive frames. Our tracker first computes dominant depth blob of the object, derived from temporal depth blob propagation and represented as clusters of superpixels[1]. Then we parse and prune the pseudo foreground blob or outliers of similar depth values with Support Vector Machines (SVM) based shape filters and a weak color-shape model. Lastly, the occlusion reasoning scheme is executed to detect occlusion which is accompanied with sharp depth changes in the object area or rapid appearance alterations with abnormal location movement. Under complete occlusion, object reacquisition module will search for possible object reappearance near the occluding outliers. The overall framework is summarized in Fig. 1.

Our main contributions are three-fold. First, we propose a novel and effective depth blob propagation method for object tracking in crowd dynamic scenes. Second, we show how the combination of the principal axis based foreground parsing scheme and the SVM-based single object shape filter alleviates drift problem, even under sharp appearance changes or low contrast between the foreground and background. Thirdly, the outlier blob localization approach and switching strategy are proposed to maintain effective searching window for object reacquisition after complete occlusion. The superior performance against the appearance-based [3, 12, 5] and detection-based tracking approaches [13, 14, 10] on well-known binocular datasets [8, 15, 16] proves the effectiveness of the proposed tracking algorithm.

[1] Depth-field superpixels are obtained by image segmentation [11] on *LUV* color space.

2 Previous Work

We compare our depth-based tracking approach with five categories of tracking methods. First, Shotton [17] shows how complex human postures can be reliably recognized in indoor environment from a single depth image captured by the Microsoft Kinect camera. Their works focus on human pose estimation that is a different application. Moreover, Kinect does not work well outside of limited field of view and under strong outdoor lighting, which is hard to be applied in outdoor crowd dynamic scenes. Second, multi-camera tracking approaches [18, 19] can tackle multiple human tracking with estimated ground plane but impose geometrically constrained field of view, thus cannot be applied on a moving platform or in dynamic scenes. Third, our explicit depth parsing is superior to highly sophisticated appearance-based trackers [1–6] since depth cue is more stable and indicative for object localization in dynamic scenes. Fourth, depth-assisted tracking methods [20–23] are very sensitive to depth noise, which results in unstable tracking performance.

Lastly, the most relevant previous works are the multi-person tracking systems [24, 8, 9], where depth cue is used as an auxiliary cue to augment object hypothesis derived from object detection. Additionally, [10, 14, 13] extract ROIs from an image by projecting the 3D points from a depth map onto the estimated 2D ground plane and in turn constrain object detection within those ROIs. Interactions between different pedestrians are explicitly modeled using long-term tracking trajectory candidates optimization and selection in a data association manner. However, [24, 8–10, 14, 13] do not study the technical feasibility to employ depth cue to lead the tracker to effectively localize and segment an object from a visually cluttered background without heavily trained object detectors. Our approach attempts to use explicit stereo depth to segment an object from depth map, which greatly leverages the robustness of depth cue under sharp object appearance changes and low object-background contrast. No object detection, ground plane estimation nor visual odometry information are utilized for tracking, which results in a much simpler system. The main focus of this paper is to develop a state-of-the-art depth-driven robust tracking method (beneficial to other cues) rather than multi-cue integration and feedback based tracking systems [20, 24, 8, 25].

3 Depth-based Figure-Ground Segmentation

We explore the robust handling of *roughly continuous uniform depth for object in space*, suggested by depth cue. The precise depth estimation of non-informative areas such as sky or ground are not required for object tracking. Thus we use the stereo algorithm developed by [7] to generate dense depth map for tracking.

3.1 Depth-based Pixel Clustering

We observe that depth information depicts the coarse representation of the object shape (e.g., pedestrian silhouette), which can be applied to segment the object

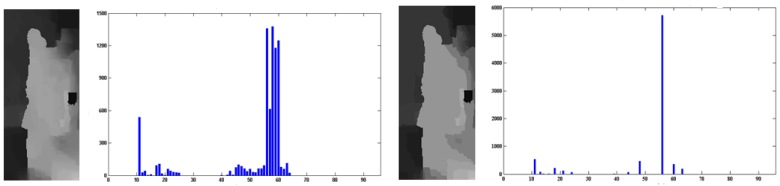

Fig. 2. Depth distributions before (**Left**) and after (**Right**) depth clustering within object region. Depth distributions are represented by depth value histograms.

from the background. In [24, 8], the dominant object depth is estimated with the median depth $d(d_i) = med_{p \in bbox_i}(p)$ within the bounding box $bbox_i$ that encompasses the object. However, this simple method is not robust without explicitly handling the object-level segmentation due to the unpredicted depth distribution of the object. In such case, the outliers can shift the dominant depth from interest region to the background. In our approach, we apply the mean shift clustering technique [11] to group pixels by considering both their spatial support and depth value, to generate reasonable depth groups and simultaneously attain the modes of each group which can more accurately indicate the dominant depth of the region. The feature space lies in the depth value of the pixel and the multivariate kernel is defined below,

$$K_{h_s,h_d}(x) = \frac{C}{h_s^2 \times h_d^p} k(\left\| \frac{x_s}{h_s} \right\|^2) k(\left\| \frac{x_d}{h_d} \right\|^2) \tag{1}$$

where p is set to 3 to increase the weight of the depth cue and C is a constant. Here we use *Normal Kernel* for both spatial dimension $\{x_s\}$ and depth cue $\{x_d\}$. A comparison result is demonstrated in Fig. 2. We can clearly see that spatial depth grouping not only make the dominant depth value statistically more distinct, but also incorporates more object-affiliated pixels into the dominant depth group, compared with the result without clustering.

3.2 Superpixel-based Segmentation with dominant depth group

Following section 3.1, we display the process of object support map generation with dominant depth group and superpixels in Fig. 3. Because there usually exists changes in object scale and location from frame to frame, we design a loopy adjustment for object bounding box where the estimated bounding box and segmentation result interplay with each other to acquire best foreground segmentation.

In detail, given the previous rectangular object box $Bbox^{t-1}$ and superpixels at t, we first compute the dominant depth group of superpixels L^t and its value D^t at t as *inliers* via $Bbox^{t-1}$. This is based on the assumption that the object normally occupies most of the previous area $Bbox^{t-1}$ in most sequences. Then we expand the box scale to get $EBbox^{t-1}$ which holds the same center of

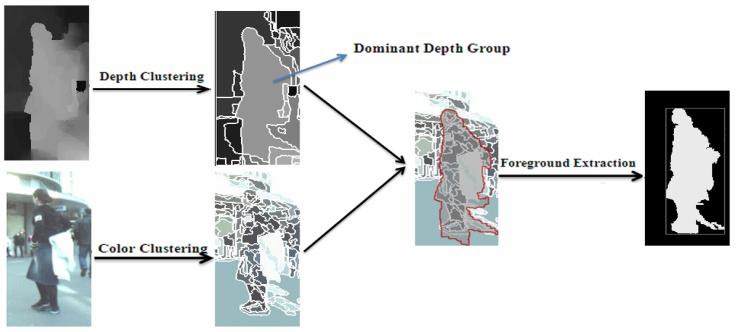

Fig. 3. The process of the depth-driven segmentation framework. Despite non-ideal depth estimation in the object boundary areas, superpixels and depth cue interact with each other to generate good object support map for foreground extraction.

$Bbox^{t-1}$ but with constant $\beta(>1)$ times larger scale (e.g., 1.2). It includes more superpixels in image proximity to verify whether they belong to the object. The confidence of a superpixel $sp_r^t \in EBbox^{t-1}, r = 1, 2, ...$ at time t, associated with the object is calculated as

$$C(sp_r^t) = \frac{\sum_{i \in sp_r^t} DD(i)}{N(sp_r^t)} \tag{2}$$

Note that $DD(i)$ makes a binary decision whether pixel i belongs to the dominant depth group (i.e., *inliers* $DD(i) = 1$ versus *outliers* $DD(i) = 0$), and $N(sp_r^t)$ is the number of pixels of sp_r^t. By combining all superpixels' confidences $\in EBbox^{t-1}$, we can obtain a confidence-based object support map and a new bounding box $Bbox_0^t$ which tightly encompasses all tested superpixels with $C(sp_r^t) > 60\%$. However it is still possible that the $EBbox(t-1)$ may not incorporate all the superpixels with high confidence to the object (e.g., the object is under-segmented). Therefore we use $Bbox_0^t$ as the initial window estimate, expand it to obtain $EBbox_0^t$ and repeat the above foreground confidence counting and mapping update to compute more accurate $Bbox_1^t$. This process is iterated n times until convergence. Finally, we set $Bbox_n^t$ as $Bbox^t$ where normally $n = 2, 3$.

In summary, we utilize the depth information to classify superpixels which preserve the inherent silhouette of the object and in turn accurately modify the coarse foreground boundary corresponding to depth discontinuity; whereas depth cue acts as a bridge to lead superpixels to find the geometric reliable object group, since depth map is computed from binocular images geometry relations.

4 Multi-object Figure Parsing via Shape and Color

Depth-based figure-ground segmentation can extract the tracked object from the background in most cases. However, outliers with similar depth values can

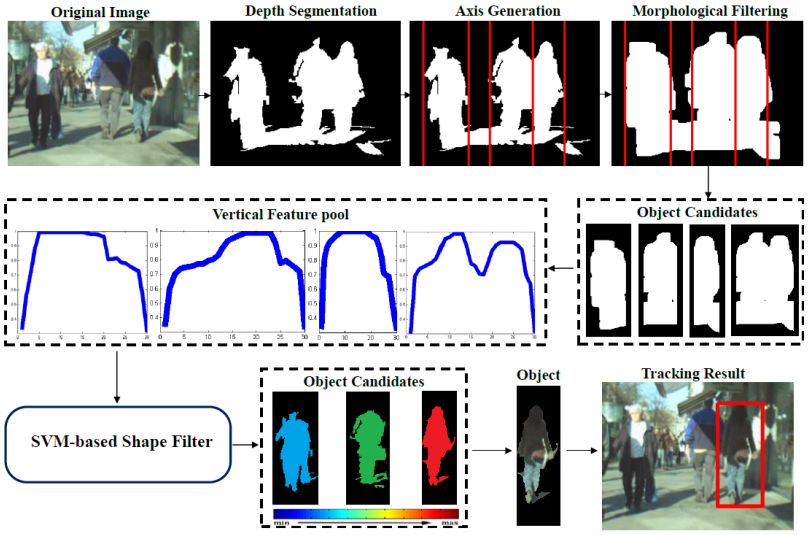

Fig. 4. The pipeline of splitting and parsing scheme for depth blob. Here we only display a part of the candidate set for better clarity. The result of similarity measure for each candidate is showed in different colors. Warmer color represents higher similarity.

contaminate the dominant depth group. To address this problem, we propose to split the depth blob into several foreground hypotheses and in turn test them via a trained SVM shape filter [26]. Hypotheses that fail to satisfy the shape feature of single objects are eliminated. Therefore, the dominant depth group is divided into a collection of single object blobs. Finally the histogram-based template matching scheme selects the best candidate with the highest joint space-color and shape similarity. The entire process is illustrated in Fig. 4.

4.1 Support Vector Machines for Shape Filtering

We assume the direction of an object's principal axis is vertical [27] which is valid in most scenarios, although more sophisticated principal-axis estimation methods can be employed. It is observed that a single object (e.g., a pedestrian or a car) retains one dominant peak if we project the foreground segmentation mask onto the vertical axis, while blobs containing multiple objects or object+background structures have multiple peaks. Motivated by the difference in vertical shape patterns between single objects and more complicated cases, we adopt Support Vector Machines to train an efficient single object detector using the vertical shape features.

There are five steps to generate the vertical shape feature: 1) Given a foreground hypothesis, its Y-coordinate centroid is first calculated as $M_y = \sum_{y=1}^{height} y \times \widetilde{H}(y) / \sum_{y=1}^{height} \widetilde{H}(y)$ where $\widetilde{H}(y)$ records the pixel number in the yth row; 2) Count the number of pixels above M_y belonging to the foreground hypothesis i

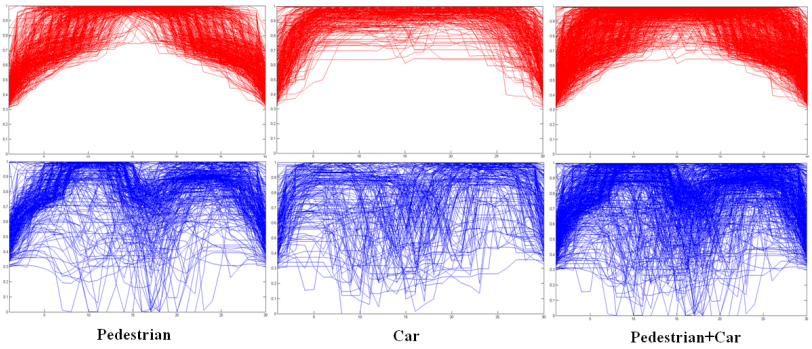

Fig. 5. Vertical shape feature of three classes. We use red curves to represent positive feature plots and blue ones for negative feature.

	Positive training samples	Negative training samples	Testing samples	Detection Accuracy
Pedestrian	432	276	108	98.15% (106/108)
Car	331	164	95	95.79% (91/95)
Pedestrian+Car	763	440	303	97.03%(294/303)

Fig. 6. SVM classification and validation accuracy.

per bin along X axis thus we obtain the histogram $HV_i(x) = \sum_{y=M_y}^{height} \widetilde{F}(x,y); x = 1,2,\ldots,w$ whose dimension w accords with the width of the hypothesis i. The function $\widetilde{F}(x,y)$ makes the binary decision whether pixel at location (x,y) belongs to foreground (i.e., *inliers* $\widetilde{F}(x,y) = 1$ versus *outliers* $\widetilde{F}(x,y) = 0$). It is necessary to constrain the projection domain because vertical shape of object's upper part is more discriminative; 3) Since hypotheses with different width have different dimensions of HV which cannot be applied in SVM, we enlarge the dimension of HV_i to EHV_i with fixed length EL by linear interpolation; 4) Apply median operator to extract median between every K successive bins EHV_i for subtracting the EHV_i to $V_i(x)$ with the fixed length L: $V_i(x) = median(EHV_i(K \times x), \ldots, EHV_i((K+1) \times x)); x = 1,2,\ldots,L$ and $K = EL/L(EL > L)$; 5) The sum of the dimension of feature vector is normalized to 1 during training. LibSVM [26] is employed in our experiment.

In our experiment, we train two separate classifiers for pedestrian C_p as well as car C_c, and one joint classifier for the union of pedestrian and car classes C_{pc}. We manually label single object blobs with different poses and scales as positive examples and the weakly connected masks of multiple objects (mainly 2 and 3 in our training dataset) and object+background as negative examples. Fig. 5 shows that the positive and negative feature distributions are significantly different. The details on the sizes of the training and testing datesets and the validation

classification accuracies are reported in Fig. 6. These three classes all achieve high accuracies of 98.15%, 95.79% and 97.03% respectively. The performance of the joint pedestrian and car classifier C_{pc} is as good as the separate classifiers so it is kept as default. The empirical evaluation of C_{pc} on 10 testing sequences shows that the model C_{pc} achieves stable and precise classification results.

Object Hypothesis Generation & Splitting: At runtime, we split the segmented depth blob vertically [2] and generate multiple single object hypotheses for SVM testing. Given a depth blob mask, we find the local minima of its vertical projection histogram as hypothesized splitting axes denoted by $\{\hat{X}_i\}$. The splitting axes serve as the boundaries of the candidates along the vertical direction. Any subregion bounded by two hypothesized splitting axes is considered as a valid hypothesis for SVM testing. After hypotheses sampling and SVM classification, single object blobs are returned for further selection. Randomly perturbing on $\{\hat{X}_i\}$ with small variance has no impact on the final performance. An example is shown in Fig. 4.

4.2 Joint Space-Color Features and Similarity

Since foreground segmentation has preserved the inherent shape information encoded by the object contour, we can fully exploit the joint appearance measurement using all channels of space, color and shape. In order to alleviate the negative impacts caused by the noisy object boundary segmentation which is common in tracking, we use *radial* spatial histograms to reflect object shape distribution and integrate color similarity measure into it. The appearance model is derived from S^{t-1}_{max} on time $(t-1)$ as A_M for tracking if there is no heavy partial or full occlusion detected. More sophisticated template updating principles are also applicable and we leave this as future work.

The calculation of space-color similarity measurement is described as follows. 1) We compute the centroid of S^{t-1}_{max} and subdivide S^{t-1}_{max} into $M = 10$ components per 36^o in radial direction using centroid as the center. 2) The RGB color histogram is calculated per bin thus we obtain $HC^{t-1}_{max}(m); m = 1, 2, ..., M$ and each $HC^{t-1}_{max}(m)$ is a vector which encodes the joint spatial and color information. 3) For each bin, we simply count the foreground pixel number and normalize it with the pixel number of S^{t-1}_{max}. Thus the spatial binned occupancy map ratios $R^{t-1}_{max}(m)$ are calculated and $\sum_m R^{t-1}_{max}(m) = 1$. 4) We perform the same process to obtain histograms $HC^t_k(m)$ for each hypothesis S^t_k. 5) We have the aggregated similarity score as

$$Similarity(S^t_k; A_M) = \sum_{m=1}^{M} R^{t-1}_{max}(m) \times S(HC^{t-1}_{max}(m), HC^t_k(m)) \qquad (3)$$

[2] In order to reduce local structure noise caused by blob fragmentation and articulation, we apply an isotropic Gaussian filtering to smooth the boundary of depth blob. In our implementation, the low-pass Gaussian filter has a size of 15×15 voxels with standard deviation set to 5.

where $S(HC_{max}^{t-1}(m), HC_k^t(m))$ is the *Bhattacharyya* coefficient between two histograms.

5 Occlusion Handling

Occlusion can be reliably detected when the dominant depth value in immediate previous object area has sharply changed. Consequently, the tracker will locate the outlier blob (e.g., other pedestrians in closer distance to camera) which results in the occlusion, and track it while actively searching for the original target which may reappear near the location of the outlier blob (**OB**) later.

Occlusion Detection: We exploit two conditions where occlusion may occur, designated by $flag_{occ} = true$. First, during tracking we monitor sharp dominant depth change inside the bounding box at the previous frame to detect occlusion. Denote the previous dominant depth as D^{t-1} in bounding box at Box^{t-1} which is parameterized using its center location $(x, y)^{t-1}$ and scales $(s_x, s_y)^{t-1}$. Occlusion status is flagged when $\left|D^{t-1} - D^t\right| > G(D^{t-1})$ where the function $G(D^{t-1})$ returns a linear ratio of D^{t-1}. Thus in the first scenario, occlusion is detected when the dominant depth value changes more than a significant percentage of itself. Second, In some challenging scenes, the target may interact with other objects with similar depth value when occlusion occurs. Therefore we need to establish more elaborated strategies to resolve the occlusion detection problem under this condition. If the joint appearance similarity score, $Similarity(S_{max}^{t-1}; S_{max}^t)$, suffers from a rapid decrease and the between-frame translation distance in the 3D coordinates (from stereo vision) is abnormally large, we consider the target occluded by the outlier blob.

Target Reacquisition: Based on the fact that the target is hidden behind the front **OB** under complete occlusion until it reappears, our tracker will plant depth seeds with same scale of the object before occlusion within and near the immediate previous object area, and run the segmentation engine to find the **OB**. Next we track this **OB** to keep an effective searching proximity for reacquiring object. Without loss of generality, we assume that $flag_{occ} = true$ starts at $(t-1)$. The tracker will record the object appearance model from S_{max}^{t-1} before occlusion and relocate it when encountering similar appearance around **OB**'s spatial occupancy. Object reacquisition runs by randomly sampling depth seeds along the boundary area of S_{max}^T in successive frames $T = t-1, t, t+1, ...$, to see if it can identify and form uniform depth blob candidates $S_k'^T$ via superpixel groups (applying the segmentation method to each seed as described in Section 3). If the best candidate's similarity score $\max_k(Similarity(S_{max}^{t-1}; S_k'^T))$ is above a certain threshold η, we relabel the depth blob as the object and reset the tracker to the normal state $flag_{occ} = false$.

6 Experiments

We first evaluate our tracker on 10 sequences (17 sub-sequences and 3000+ frames) from well-known binocular datasets [8, 15, 16]. These sequences include

	Bahnhof				Jelmoli			Loe.	Sunny Day		Vicon1	Vicon2	Drive1	Drive2	Drive3	Drive4	
index	a	b	c	d	a	b	c	a	a	b	a	a	a	a	a	a	b
Frames	39	51	51	51	188	206	206	91	354	354	221	301	159	200	201	221	185
DDT	7	5	4	11	7	5	7	36	8	9	15	25	4	3	6	11	9
SPT	22	8	6	11	177	25	9	39	59	38	160	103	150	168	70	182	11
TLD	46	5	31	45	160	2	7	69	99	66	230	130	44	118	298	7	51
HT	32	122	113	14	88	33	34	36	143	144	267	188	185	208	96	23	125

Fig. 7. Tracking results. The numbers for each tracker denote average errors of center location in pixels. The best and second best results are shown in red and blue for each sub-sequence.

	Bahnhof				Jelmoli			Loe.	Sunny Day		Vicon1	Vicon2	Drive1	Drive2	Drive3	Drive4	
index	a	b	c	d	a	b	c	a	a	b	a	a	a	a	a	a	b
Frames	39	51	51	51	188	206	206	91	354	354	221	301	159	200	201	221	185
Mean	7	5	4	11	7	5	7	36	8	9	15	25	4	3	6	11	9
Std.	4.31	5.34	7.3	6.99	3.47	4.81	2.21	43.4	9.70	10.9	15.4	20.7	6.98	3.00	6.40	3.00	14.1
Max	21	19	17	23	17	19	13	@82	30	28	52	63	14	12	20	13	39
E/S(%)	6.77	7.85	5.40	6.33	7.42	7.90	8.93	16.2	8.65	9.30	9.18	8.36	8.80	10.6	7.57	10.9	8.11
(E/S<10%)	82.1	74.6	86.3	84.3	80.3	73.3	79.1	79.1	75.1	65.5	75.6	79.0	76.7	68.5	77.6	53.2	71.4

Fig. 8. Quantitative evaluation on tracking accuracy (in pixels and error-to-scale ratio, E/S(%); percentage of frames $E/S < 10\%$). @82 means the tracker lose track of the object at frame 82. The sequences whose $(E/S < 10\%)/F > 70\%$ is highlighted in red.

most challenging factors in visual tracking: dynamic background, complete occlusion, fast movement, large variation in pose and scale, shape deformation and distortion. The quantitative evaluations of our tracker (**DDT**), superpixel tracking [3] (**SPT**), P-N learning [12] (**TLD**), hough-based tracking [5] (**HT**) are presented in Fig. 7. The codes of all competing methods are publicly available [3]. Fig. 8 shows statistics on mean, maximum distance and standard derivation (std.) of the error $|\Delta|$ from the tracked object center to manually annotated ground truth center in pixels and the mean of error-to-scale ratio (i.e., E/S(%) $= |\Delta|/sqrt(S_x^2 + S_y^2)$ where S_x and S_y are the lengths of an object annotated bounding box in x or y coordinate respectively.) of 3000+ labeled frames.

Our absolute object tracking errors (in pixels) are highly competitive or better than most recent results [3, 12, 5], which strongly proves the need to use depth cue in dynamic scenes. Although the maximum error in some sequences are relatively high due to depth noise, the overall tracking performance is stable and outstanding with low mean of error-to-scale ratio. Some comparison of tracking results are shown in Fig. 9.

In recent years, the depth-assisted systems [13, 14, 10] achieve better detection recall rates on the same datasets [8] and also outperform in other scenarios, so we choose to use the numerical results from [13, 14, 10] and compare our recall rate (successful tracked objects/total annotations), derived from the same evaluation criteria in [8, 13, 14, 10]), with the defined best recall rates (false posi-

[3] **SPT**:http://ice.dlut.edu.cn/lu/iccv_spt_webpage/iccv_spt.htm.
TLD:http://info.ee.surrey.ac.uk/Personal/Z.Kalal/tld.html.
HT:http://lrs.icg.tugraz.at/research/houghtrack/index.php

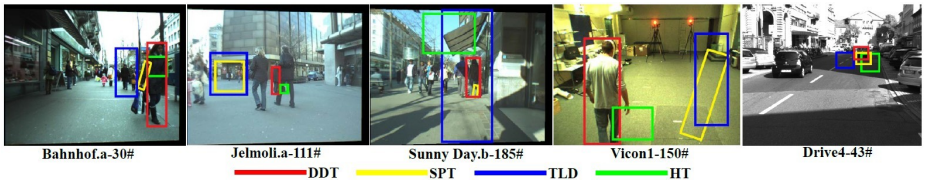

Fig. 9. Tracking results. The results by our tracker, superpixel tracking [3], P-N learning [12], hough-based tracking [5] are represented by red, yellow, blue and green rectangles.

	Bahnhof	Sunny Day	Jelmoli
DDT	**0.89** (1775/1997)	**0.87** (874/1002)	**0.80** (955/1187)
Mitzel'11	**0.70**	**0.82**	0.60
Bansal'10	0.48	0.50	**0.65**
Bajracharya'09	0.47	0.71	0.46

Fig. 10. The table shows the recall rate (successful tracked objects/total annotations) for our tracker and three competing methods [13, 14, 10] (false positives/image = 0.5). The best and second best results are displayed in red and blue for each sequence.

tives/image=0.5) in their ROC curves. Here we use Ground-truth [8] to initialize our tracker's location window on the first frame where object appears, which is also used in [13, 14, 10]. For the rest of the testing sequences, we initialize the object location window manually. All comparative experiments are conducted using the same initial location. Since our method is not based on detection, we cannot further provide ROC curve analysis and just take part of the annotations (1997 from 5193 of Seq. Bahnhof, 1002 from 1828 of Seq. Sunny Day and 1187 from 2697 of Seq. Bahnhof) for testing. As can be seen in Fig. 10, our tracker achieves better results with higher recall rate in all three datasets.

Self-recovery from figure corruption: In Fig. 11, due to large shadow or untextured area caused by strong sunlight, depth cue frequently fails to provide valid uniform depth value within object area, which results in inaccurate figure-ground segmentation (e.g., frames 36, 89, 168). Even under this condition, depth cue can still lead the depth-driven tracker to reliably locate part of the object with small amount of outliers, while saving the previous appearance model when the scale changes rapidly. When the tracker encounters more desirable frames (e.g., frames 45, 95, 171) with uniform depth in object region later, the figure can recover from inaccurate segmentation in previous frames by crawling along uniform depth blob of the object.

Comparison of Depth and Appearance Cue: In Fig. 12, we show the performance of depth and appearance cues in support of the tracker to locate the object. To accurately depict the fluctuation trend of appearance cue, we take ground truth of the object to calculate the similarity value using the method described in section 4.2. We can see that when the appearance model sharply changes (e.g., similarity measure value between two frame below 0.7), the depth

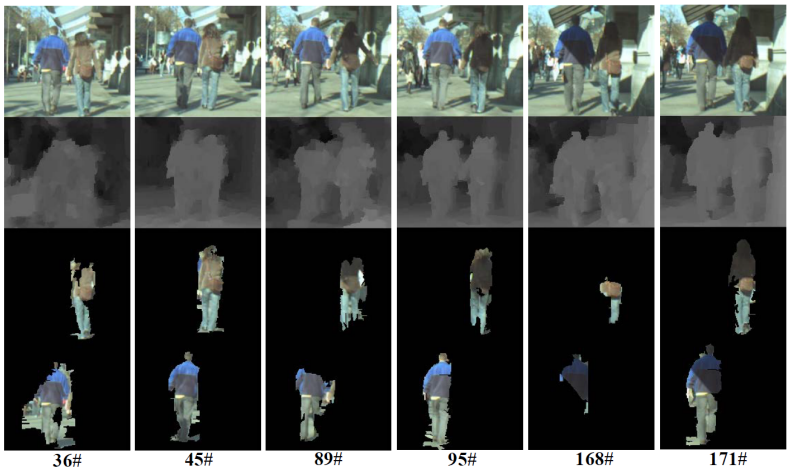

Fig. 11. Tracking results of Sunny Day sequence. The original object area and its surroundings, the corresponding depth map, the segmentation results of the "woman" object and the "man" object, are respectively displayed in the vertical direction for each frame. Successful object tracking and segmentation recovery from severe depth noise are demonstrated.

cue consistently maintains smooth variation, which is more favorable for robust tracking. The key frames with rapid appearance changes are shown in Bottom-left of Fig. 12 (also highlighted by green ellipses in Top graph). The key frames in the bottom-right of Fig. 12 (highlighted by purple ellipses in Top graph), most parts of the object are occluded so that both depth and appearance cues have abrupt changes, which indicates occlusion occurrence. Then the tracker switch the tracking target to the **OB** encompassed by blue bounding box while continuously searching in the proximity region until object reappears.

Failure Case Analysis: The tracker could lose the object when more complex occlusion patterns (e.g., multiple spatially-correlated **OB**s) occlude the object and prevent it from reappearing close enough to the **OB**. In Fig. 13, when the object is mostly visible at frame 245, it is out of our reacquisition searching range, interleaved by another **OB** in between. This can be improved by designing elaborated **OB**-**OB** and object switching strategy (e.g., data association model).

7 Conclusion

In this paper, we present a robust tracker using explicit stereo depth with occlusion handling for tracking a single object in dynamic and crowd scenes. We successfully validate the proposed method using several stereo video sequences under various challenging conditions (indoor/outdoor) such as occlusions, illumination and appearance changes, etc. For future work, we will explore more sophisticated online appearance models [29, 3] and multi-cue integration systems.

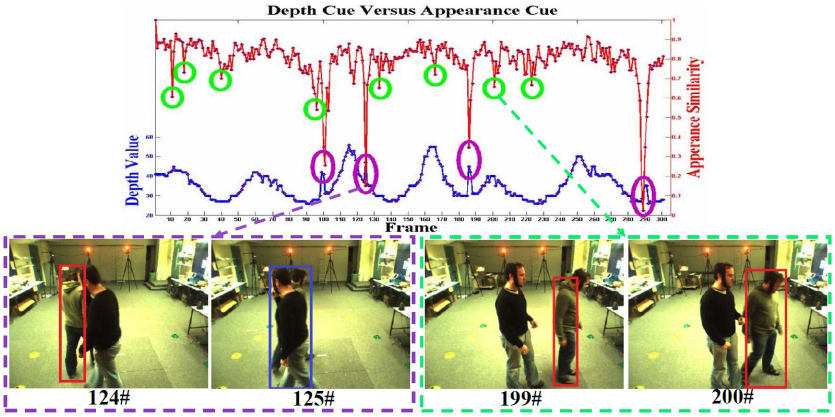

Fig. 12. The top graph shows the performance of both depth cue and appearance cue in support of tracker to locate an object [28]. Blue curve indicates the change of dominant depth value of foreground region captured by our tracker along a 301 frame sequence and the red curve depicts the fluctuation of object appearance.

Fig. 13. Failure case due to more than one **OB** (in blue box) occluding the object successively at frame 241 so that the tracker cannot regain the target around the first **OB** at frame 245.

Acknowledgement. The work was supported by the National Natural Science Foundation of China (61170179), the Special Research Fund for the Doctoral Program of Higher Education of China under Project (20110121110033), and Xiamen Science & Technology Planning Project Fund (3502Z20116005) of China. For more details, please correspond with Prof. Hanzi Wang.

References

1. Ren, X., Malik, J.: Tracking as Repeated Figure/Ground Segmentation. CVPR (2007)
2. Lu, L., Hager, G.: A Nonparametric Treatment for Location Segmentation Based Visual Tracking. CVPR (2007)
3. S. Wang, H. Lu, F.Y., Yang, M.: Superpixel Tracking. ICCV (2011)
4. Liu, B., Huang, J., Yang, L., Kulikowsk, C.: Robust tracking using local sparse appearance model and k-selection. In: CVPR. (2011)
5. M. Godec, P.M.R., Bischof, H.: Hough-based Tracking of Non-Rigid Objects. ICCV (2011)

6. Prisacariu, V., Reid, I.: Nonlinear shape manifolds as shape priors in level set segmentation and tracking. In: CVPR. (2011)
7. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Belief Propagation in Early Vision. CVPR (2005)
8. Ess, A., Leibe, B., Schindler, K., Gool, L.: Robust Multi-person Tracking from a Mobile Platform. IEEE Trans. PAMI (2009)
9. Schindler, K., Ess, A., Leibe, B., Gool, L.V.: Automatic detection and tracking of pedestrians from a moving stereo rig. ISPRS (2010)
10. Mitzel, D., Leibe, B.: Real-time multi-person tracking with detector assisted structure propagation. CORP (2011)
11. Comaniciu, D., Meer, P.: Mean shift A robust approach toward feature space analysis. IEEE Trans. PAMI (2003)
12. Z. Kalal, J.M., Mikolajczyk, K.: P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints. CVPR (2010)
13. Bajracharya, M., Moghaddam, B., Howard, A., S.Brennan, Matthies, L.: A fast stereo-based system for detecting and tracking pedestrians from a moving vehicle. IJRS (2009)
14. Bansal, M., S. H. Jung, B.M., Eledath, J., Sawhney, H.S.: A real-time pedestrian detection system based on structure and appearance classification. ICRA (2010)
15. Geiger, A., Roser, M., Urtasun, R.: Efficient large-scale stereo matching. In: ACCV. (2010)
16. P. Kelly, N.O., Smeaton, A.: A Framework for Evaluating Stereo-Based Pedestrian Detection Techniques. IEEE Transactions on Circuits and Systems for Video Technology, Volume 18, Issue 8, Pages 1163-1167 (2008)
17. Shotton, J., Fitzgibbon, A., et al.: Real-Time Human Pose Recognition in Parts from Single Depth Images. CVPR (2011)
18. Kim, K., Davis, L.S.: Multi-camera Tracking and Segmentation of Occluded People on Ground Plane Filtering. ECCV (2006)
19. Khan, S., Shah, M.: Tracking multiple occluding people by localizing on multiple scene planes. PAMI (2009)
20. Darrell, T., Gordon, G., Harville, M., Woodfill, J.: Integrated Person Tracking Using Stereo, Color, and Pattern Detection. IJCV (2000)
21. Harville, M., Li, D.: Fast, integrated person tracking and activity recognition with plan-view templates from a single stereo camera. (2004)
22. Bleiweiss, A., Werman, M.: Fusing Time-of-Flight Depth and Color for Real-Time Segmentation and Tracking. Dynamic 3D Imaging (2009)
23. Wang, L., Zhang, C., Yang, R.: TofCut: Towards Robust Real-time Foreground Extraction Using a Time-of-Flight Camera. 3DPVT (2010)
24. Ess, A., Leibe, B., Gool, L.V.: Depth and Appearance for Mobile Scene Analysis. ICCV (2007)
25. Gavrila, D.M., Munder, S.: Multi-cue pedestrian detection and tracking from a moving vehicle. IJCV (73:41-59, 2007)
26. Chang, C., Lin, C.: Libsvm: a library for support vector machines. ACM Trans. on Intelligent Systems and Technology (2011)
27. Haritaoglu, I., Harwood, D., Davis, L.: Real-Time Surveillance of People and Their Activities. IEEE Trans. PAMI (2000)
28. Kitt, B., Geiger, A., Lategahn, H.: Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. In: IEEE IVS. (2010)
29. Lu, L., Hager, G.: Dynamic Background/Foreground Segmentation From Images and Videos using Random Patches. NIPS (2006)