Towards Generic 3D Tracking in RGBD Videos: Benchmark and Baseline

Jinyu Yang[1,2], Zhongqun Zhang[2], Zhe Li[1], Hyung Jin Chang[2], Aleš Leonardis[2], and Feng Zheng[1⋆]

[1] Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China
[2] University of Birmingham, Birmingham, U.K.

Table 1. Description of attributes in our dataset.

Attribute	ID	Description
Background Clutter	BC	Background has similar colors with the object.
Camera Motion	CM	There are abrupt motions of the camera.
Deformation	DF	The target object is deformable.
In-plane Rotation	IR	Target rotates in-plane.
Illumination Variation	IV	The illumination is too low or high or varies.
Out-of-plane Rotation	OR	Target rotates out-of-plane.
Point Cloud Sparsity	PCS	Point clouds are too few to distinguish the object.
Similar Targets	ST	There are objects similar to the target in the scene.
Target Loss	TL	The target is fully occluded or out-of-view.

A Attribute Description

We give detailed description of each attribute in Table 1.

B Analysis on 3D Annotation Accuracy

To ensure high-quality annotation, we test the similarity between the projected 2D BBoxes from our 3D annotation and the manually annotated 2D BBoxes. In the projection process, we project the 3D BBox to a 2D plane to give 2D-level annotations, *i.e.*, axis-aligned BBox. The projection from 3D to 2D BBox is implemented by finding the minimum point set and forming a bounding rectangle given 8 corner points. Following the principle that the 2D BBox will tightly fit the 3D BBox according to the associations, we generate the 2D BBoxes from the projection of 3D BBox. We manually annotate 10% of the data randomly with 2D BBoxes, which is used to validate 3D annotation accuracy. Then the projected BBoxes will be compared with the manually annotated ones. The visualisation of comparison is shown in Fig. 1. We calculate the average projection error

⋆ Corresponding author.

and average IoU between them, which is 12.2 pixels and 66%, indicating that our annotated 3D BBoxes are reliable. Besides, the projection enables multiple evaluations and mixed-use on the proposed dataset, which will be discussed in Sec. E.2.

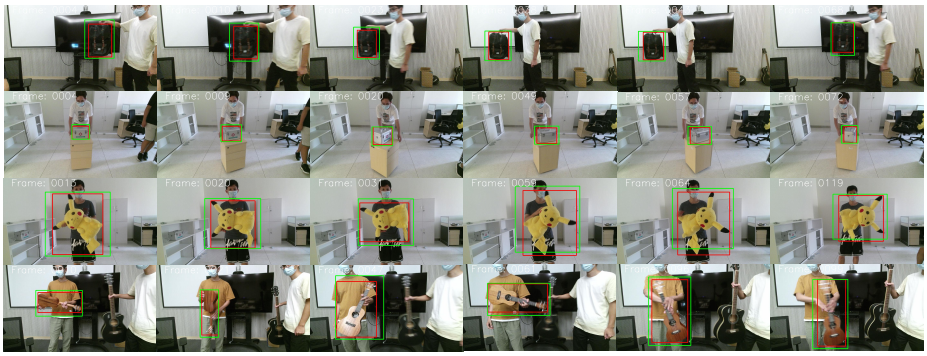

Fig. 1. Qualitative examples of projected 2D BBoxes from 3D annotation (Green) and manually annotated 2D BBoxes (Red).

C Evaluation Protocols

Here we give details of the 3D IoU calculation, which is used as one of our evaluation protocols. According to [3], the IoU measure for general 3D-oriented boxes is based on the Sutherland-Hodgman Polygon clipping algorithm. For two 3D boxes, we firstly transform both boxes using the inverse transformation, after which one box will be axis-aligned and centered around the origin. Then, each face is clipped as the convex polygon between the predicted box and the ground truth box according to the polygon clipping algorithm. Finally, the IoU is computed from the volume of the intersection and the volume of the union of two boxes by swapping the two boxes, as used in [1].

D More Details of the Proposed Method

Backbones. The network architecture is shown in Fig. 2. The model of Sparse 3D CNN is a 3D U-Net structure, in which the convolution and up-convolution are implemented as sparse convolution [2]. As shown in the left part of Fig. 2, the parameters of each convolution layer are the number of input pixels and the size of output features respectively. In the right part of Fig. 2, Pointnet++ encoder-decoder takes as input a point cloud to generate dense features. We use four Set Abstraction (SA) layers [8] to downsample and encode the point clouds. We also leverage three Feature Propogation (FP) layers to interpolate

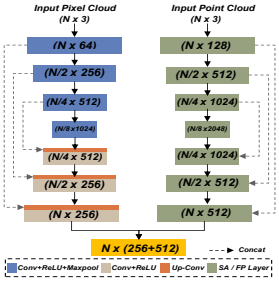

Fig. 2. The network architecture of our backbone, consisting of Sparse 3D CNN and Pointnet++.

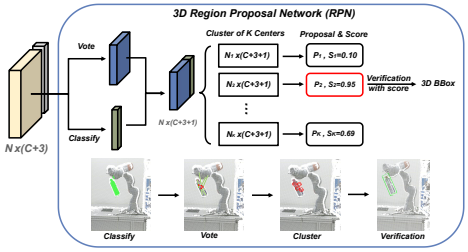

Fig. 3. Region proposal network (RPN) module in our architecture.

and decode features. The parameters of each layer are also the number of input points and the size of output features respectively. Finally, the pixel-wise features and point-wise features are concatenated together and fed into next module.

RPN module. The Fig. 3 illustrates the architecture of the VoteNet-based RPN module. A point-wise MLP (256, 256, 3+256) is applied to the fused feature for object center voting. In addition, another MLP (256, 256, 1) is used to predict a classification score (target or background) for each point. After that, the predicted vote centers and scores are concatenated together and then clustered into k groups through the furthest point sampling and the ball query. Finally, a mini-PointNet is used to produce the proposals and proposal-wise scores of each group. The proposals and scores are supervised with our annotated 3D BBox and the 3D IoU (mentioned in Sec. C) respectively.

E Experiments

E.1 More Visualised Results

Fig. 4 shows the additional results of the comparison to finetuned P2B [9]. These results demonstrate that our method is more robust to several difficulties. The first sequence shows our TrackIt3D's robustness to background clutter. The 2^{nd} and 3^{rd} sequences show that our method can well handle the problem of out-of-plane rotation, thanks to the freely rotated 3D BBox which can better describe the objects compared to the rotation-unable P2B. In the last sequence, our method can track the turtlebot under fast motion and camera motion, demonstrating the effectiveness of our proposed model compared to P2B_ft.

E.2 Extension on RGBD Tracking

With the projection from 3D BBox to 2D BBox, we can evaluate existing RGBD trackers on our proposed dataset. Fig. 5 shows the precision and success plots of state-of-the-art trackers. Here we use the projected 2D BBoxes as

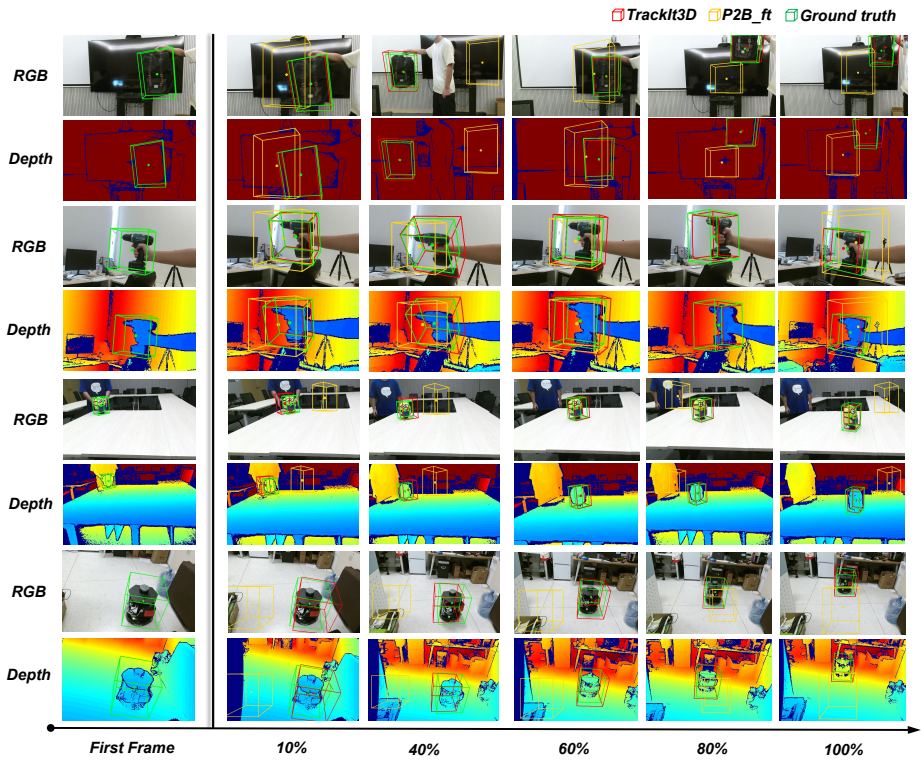

Fig. 4. More qualitative results of our baseline *TrackIt3D* compared with the fine-tuned *P2B*.

the groundtruth. The precision plots compute the Euclidean distance between target center location and labeled groundtruth position of the current frame. Different trackers are ranked with this metric on a threshold (20 pixels). Since the precision metric is sensitive to target size and image resolution, we use the normalised precision [7]. With the normalised precision metric, we rank precision scores using the Area-Under-Curve (AUC) between 0 to 0.5. The success plots calculate the Intersection-over-Union (IoU) between predicted BBox and groundtruth BBox. The tracking methods are ranked using the AUC between 0 to 1. Here we test RGBD state-of-the-art trackers on this dataset, including high-performance trackers from VOT-RGBD challenge, *i.e.*, DRefine [6], iiau_rgbd [6], SLMD [6], DDiMP [4], Siam_LTD [4], and ATCAIS [5], and advanced RGBD trackers, *i.e.*, DeT [11], TSDM [12], and DAL [10]. As shown, state-of-the-art trackers can achieve 60% to 70% on precision and success, indicating that our dataset is less challenging on 2D tasks if we only "track the object on the plane". Note that existing state-of-the-arts only obtain precision and success scores of

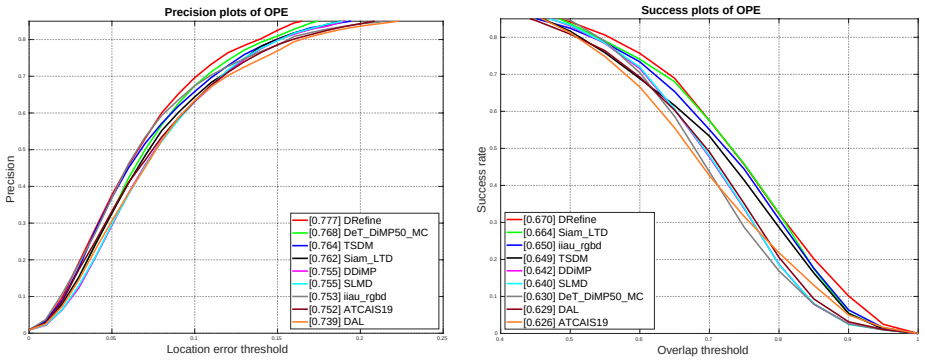

Fig. 5. Precision and success plots of evaluated trackers on our dataset with 2D settings. Compared trackers: DeT [11], TSDM [12], DAL [10], DRefine [6], iiau_rgbd [6], SLMD [6], DDiMP [4], Siam_LTD [4], ATCAIS [5].

lower than 30% on our proposed 3D task. Therefore, it is more difficult for a tracker to predict target description in 3D scenes rather than 2D BBoxes.

References

1. Ahmadyan, A., Zhang, L., Ablavatski, A., Wei, J., Grundmann, M.: Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7822–7831 (2021)
2. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3075–3084 (2019)
3. Ericson, C.: Real-time collision detection. Crc Press (2004)
4. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Kamarainen, J.K., Čehovin Zajc, L., Danelljan, M., Lukezic, A., Drbohlav, O., He, L., Zhang, Y., Yan, S., Yang, J., Fernandez, G., et al.: The eighth visual object tracking vot2020 challenge results (2020)
5. Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Pflugfelder, R., Kamarainen, J.K., Cehovin Zajc, L., Drbohlav, O., Lukezic, A., Berg, A., et al.: The seventh visual object tracking vot2019 challenge results. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (2019)
6. Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Pflugfelder, R., Kamarainen, J.K., Chang, H.J., Danelljan, M., Cehovin, L., Lukezic, A., et al.: The ninth visual object tracking vot2021 challenge results. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2711–2738 (2021)
7. Muller, M., Bibi, A., Giancola, S., Alsubaihi, S., Ghanem, B.: Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 300–317 (2018)
8. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. arXiv preprint arXiv:1706.02413 (2017)

9. Qi, H., Feng, C., Cao, Z., Zhao, F., Xiao, Y.: P2b: Point-to-box network for 3d object tracking in point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
10. Qian, Y., Yan, S., Lukežič, A., Kristan, M., Kämäräinen, J.K., Matas, J.: DAL : A deep depth-aware long-term tracker. In: International Conference on Pattern Recognition (ICPR) (2020)
11. Yan, S., Yang, J., Kapyla, J., Zheng, F., Leonardis, A., Kamarainen, J.K.: Depthtrack: Unveiling the power of rgbd tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10725–10733 (2021)
12. Zhao, P., Liu, Q., Wang, W., Guo, Q.: Tsdm: Tracking by siamrpn++ with a depth-refiner and a mask-generator. In: 2020 25th International Conference on Pattern Recognition (ICPR). pp. 670–676. IEEE (2021)