MessyTable: Instance Association in Multiple Camera Views – Supplementary Materials –

Zhongang Cai[1] *, Junzhe Zhang[1,2] *, Daxuan Ren[1,2], Cunjun Yu[1], Haiyu Zhao[1], Shuai Yi[1], Chai Kiat Yeo[2], and Chen Change Loy[2]

[1] SenseTime Research
{caizhongang,yucunjun,zhaohaiyu,yishuai}@sensetime.com
[2] Nanyang Technological University
{junzhe001,daxuan001}@e.ntu.edu.sg,{asckyeo,ccloy}@ntu.edu.sg

1 Content Summary

In the supplementary materials, we provide additonal details on:

- data collection procedure;
- data annotation procedure;
- full list of the 120 classes of objects;
- example scenes of three difficulty levels: Easy, Medium, and Hard;
- statistics of MessyTable and the three datasets evaluated in Section 5.4;
- framework;
- proposed metric IPAA;
- baselines

2 Additional Details on Data Collection

We gather a team of 10 people for data collection, we refer to them as data collectors. We define the term "setup" and "scene" as follows: a setup is an arrangement of nine cameras. The camera poses are randomly set for a setup and are reset for subsequent setups. A scene is an arrangement of all objects on the table: a random set of objects are being placed on the table. These objects are then cleared from the table and replaced with a new random set of objects for subsequent scenes. With each setup, each camera captures one photo for each scene; a total of 10 scenes are collected for each setup.

2.1 Setup

Camera Poses and Extrinsic Calibration For each setup, cameras poses, except camera #1 that provides a bird's eye view of the scene, are varied. Certain camera poses are deliberately arranged to be very near the table surface, to

* indicates equal contribution.

collect images of an incomplete scene. A calibration board, with six large ArUco [3, 7] markers are then placed on the table, at a position that is visible to all cameras. The detected marker corners are used to compute the transformation matrix from the board frame to the camera frame by solving the the perspective-n-points problem [1].

Lighting Conditions Variations in lighting often severely affect the performances of visual algorithms. Data augmentation [9] and artificially generated shadows [11] can be unrealistic. Hence, we combine fixed light sources with mobile studio lighting kits to add lighting variations to the dataset such as different light directions and intensity, shadows, and reflective materials. The lighting is adjusted for every setup.

2.2 Scene

For object placements, we only provide vague instructions to the data collectors about the approximate numbers of objects to be used for Easy, Medium, and Hard scenes respectively; the data collectors make their own decisions at choosing a set of objects and the pattern to place the objects on the table. Hence, we ensure that the object placements resemble the in-the-wild arrangements as much as possible.

For backgrounds, we include baskets and cardboard boxes during data capturing. They serve various purposes, including as occlusion, as platforms for other objects, etc. We also have coasters, placemats, and tablecloths underneath each scene which come in different sizes, patterns, colors, and textures, and are commonly found in natural scenes.

3 Additional Details on Data Annotation

The interactive tool we design for the association stage is shown in Figure 1. By selecting bounding boxes, these bounding boxes are assigned the same instance ID. The tool is designed with the following features to increase efficiency and to minimize errors:

Irrelevant Bounding Box Filtering Once a bounding box is selected (by clicking on it) in any view, only the bounding boxes of the same class or similar classes will remain displayed in other views. It is worth noting that we choose to keep similar classes, in addition to the same class, because the labels from the classification stage can be erroneous (a object is wrongly annotated with a similar class to the true class). Classes are considered to be similar based on their categories (the grouping is listed in Table 1).

Classification Annotation Verification The tool checks if the bounding boxes with the same instance ID have the same class labels. It notifies annotators if any disagreement is detected, and performs automatic correction based on majority voting of the class label amongst nine views, each annotated independently in the classification stage.

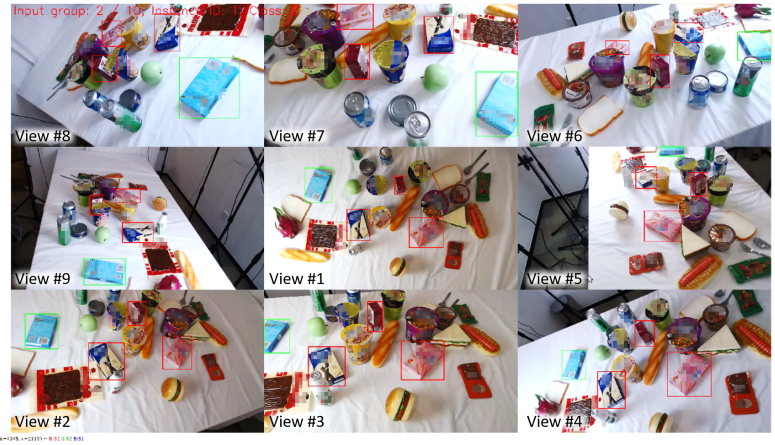

Fig. 1: The user interface of the interactive tool. The views are arranged according to the actual camera locations. The green bounding boxes are currently selected to be assigned the same instance ID. The red bounding boxes have similar class labels. The rest of bounding boxes are not displayed. The brand names are pixelated in all illustrations.

Table 1: Grouping of classes used in the association annotation stage to accelerate the annotation by filtering out irrelevant bounding boxes

Group	Class	Description
A	1-10	bottled drinks
B	11-19	cupped food
C	20-30	canned food
D	31-41	boxed food
E	42-50	vacuum-packed food
F	51-60	puffed food
G	61-77	fruits
H	78-83	vegetables
I	84-96	staples
J	97-100	utensils
K	101-107	bowls & plates
L	108-115	cups
M	116-120	drink glasses

4 Full List of 120 Object Classes

Fig. 2: The full list of the 120 classes of objects. The objects are commonly found on a table in real life. They have a wide variety of sizes, colors, textures, and materials. Supermarket merchandise: 1-60; agricultural products: 61-83; bakery products: 84-96; dining wares: 97-120. Note that highly realistic food models are used for class 61-96 as the actual food is perishable, making it not suitable for data collection which spans over a few months

5 Example Scenes

Fig. 3: Example scenes in all nine views. (a) An Easy scene with 19 objects. (b)A Medium scene with 27 objects. (c)A Hard scene with 56 objects. Harder scenes have more object instances, more severe occlusion, and more similar/identical objects. Only part of the scene is visible in some camera poses

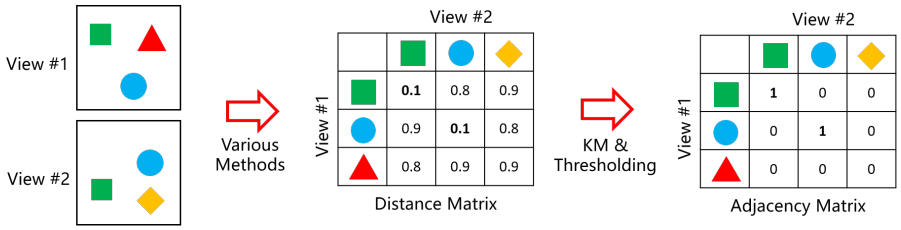

Fig. 4: The general framework for instance association in a multi-camera setting. In this example, the red triangle is only visible in View #1 and the yellow diamond is only visible in View #2. All methods we explain in the main paper essentially compute pair-wise distances between instances. KM stands for Kuhn-Munkres algorithm, which globally optimizes the matches such that the total loss (the sum of distances of matched pairs) is the minimum. An additional thresholding step further rejects matches with large distances

6 Additional Statistics of MessyTable and Other Datasets

Table 2 shows the additional statistics of MessTable and the three datasets that were evaluated in Section 5.4.

Table 2: Comparison with other multi-camera datasets. MessyTable is the largest in all aspects below.

Datasets	Classes	Cameras	Setups	Scenes	Images	BBoxes	Instances
MPII MK	9	4	2	33	132	1,051	6-10
EPFL MVMC	3	6	1	240	1,440	4,081	5-9
WILDTRACK	1	7	1	400	2,800	42,707	13-40
MessyTable	**120**	**9**	**567**	**5,579**	**50,211**	**1,219,240**	**6-73**

7 Additional Details on the Framework

As shown in Figure 4, all baselines discussed in the main paper are essentially different ways to compute the pair-wise distances. Homographic projection uses the pixel distance between two sets of projected points; SIFT uses the chi-square distance between two visual bag of words representations; MatchNet and DeepCompare use metric networks to compute the similarity between extracted feature vectors; DeepDesc, TripletNet, and ASNet use L2 distance; Epipolar soft constraint uses pixel distance between a bounding box center point and an epipolar line.

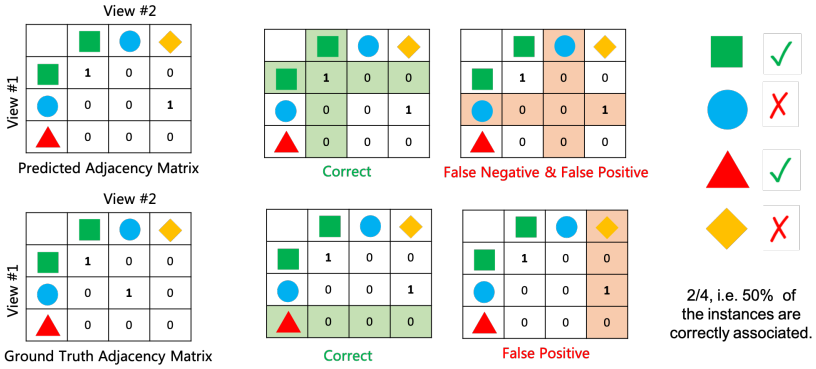

Fig. 5: Computation of the percentage of correctly associated objects in an image pair. The predicted adjacency matrix (Section 7) is compared against the ground truth for each object present in either of the two images. IPAA-X is the fraction of image pairs that have no less than X% of objects associated correctly

8 Additional Details on the Proposed Metric: Image-pair Association Accuracy (IPAA)

The motivation for IPAA is to gauge performance at the image-pair level whereas AP and FPR-95 gauge performance at the instance-pair level: AP and FPR-95 evaluate the matching score (confidence score) of each instance pair against its ground truths (0 or 1), but do not directly provide insights of the matching quality of an image pair, which contain many instance pairs. In contrast, IPAA is computed as the fraction of image pairs with no less than X% of the objects associated correctly (written as IPAA-X). The computation of the percentage of correctly associated objects for each image pair is shown in Figure 5.

9 Additional Details on Baselines

This section provides more details on baselines. These details are excluded in the main paper due to space constraint, but they offer important insights on the instance association problem.

9.1 Additional Results on Zoom-out Ratio

By including surrounding information, the key hyperparameter for our baseline ASNet is the zoom-out ratio. We also conduct experiments on different zoom-out ratios. It shows that including surrounding information significantly improves the association performance (compared to that when zoom-out ratio = 1). We simply choose the zoom-out ratio to be 2 as the performance is not sensitive to the value of zoom-out ratio in the range [1.2, 2.4]. However, as the zoom-out

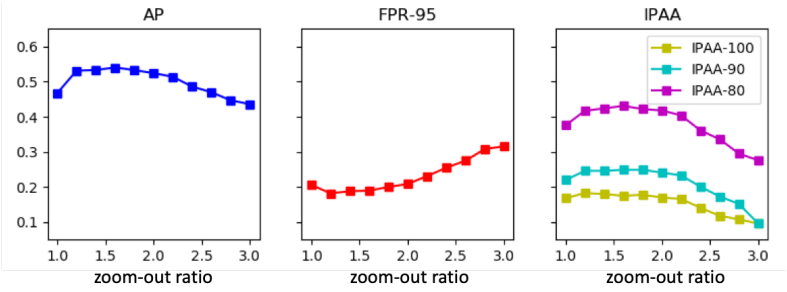

Fig. 6: Performance of ASNet is not sensitive to the value of zoom-out ratio in the range [1.2, 2.2], after which it drops rapidly

Table 3: Instance association performance of ASNet using detected bounding boxes. The instance association performance suffers from imperfect bounding boxes generated by detectors compared to ground truth bounding boxes. The performance deteriorates as the detectors become weaker

Detector	Detection mAP↑	IPAA-100↑	IPAA-90↑	IPAA-80↑
GT Bounding Box	**1.0**	**0.170**	**0.241**	**0.418**
Cascade Faster R-CNN r101	0.797	0.153	0.212	0.388
Cascade Faster R-CNN r50	0.772	0.141	0.198	0.366
Faster R-CNN r101	0.756	0.120	0.165	0.326
Faster R-CNN r50	0.722	0.097	0.135	0.283

ratio increases beyond 2.4, the performance starts to decline. We argue that even though a larger zoom-out ratio could include more surrounding area, the model is unable to extract an effective embedding for the surrounding features. This can be a direction for future research.

9.2 More Details on Using Bounding Boxes from Detectors

We also evaluate our trained ASNet model on the test set where the bounding boxes are generated by detectors, instead of the ground truth bounding boxes. These detected bounding boxes suffer from false positive (false detection), false negative (missed detection), and imperfect localization and dimension.

It is worth noting that the detected bounding boxes undergo post-processing to obtain instance IDs from the ground truth. For a given image, bipartite matching is performed between the detected bounding boxes and the ground truth bounding boxes based on pair-wise IoUs. The matched detected bounding boxes are assigned the instance IDs of the ground truth bounding boxes, whereas the unmatched detected bounding boxes are assigned unique instance IDs.

The results are collated in Table 3. Instance association itself is challenging, let alone combining it with a detection stage. The weaker the detection model

Fig. 7: Visualization of cases where both appearance features and surrounding features combined are insufficient for instance association. In this regard, the soft epipolar constraint is necessary as it assigns the geometrically infeasible pair (i.e., false pair) a larger distance

used as the upstream, the worse the association performance gets. We point out that joint optimization of the detection and the association stage can be a direction for future research.

9.3 Additional Visualization of Scenes Where Geometric Cues Are Necessary

Figure 7 visualizes the scenes where both the appearance features and the surrounding features are similar for different object instances. In this scenario, geometric cues are particularly helpful as they give penalty to the geometrically infeasible pair (i.e., false pair), hence making the overall distance of the false pair larger than that of the true pair.

9.4 Additional Results from Structure from Motion Baseline

Structure from Motion(SfM) can be used to generate 3D structure from multiple views [4,12]. The 3D structure can be trivially used for instance association from multiple views as pixel correspondences are known. However, an inherent limitation of SfM is that only the intersection of cameras' views can be reconstructed whereas instance association from multiple views should cover the union instead. Besides, SfM is sensitive to repetitive patterns, reflective, and textureless surfaces [5]. We apply three state-of-the-art SfM engines, ColMap [8], OpenMVG [6], and Theia [10], on the scenes of MessyTable. The first two are unable to reach convergence whereas Theia gives incomplete reconstruction results, shown in Figure 8.

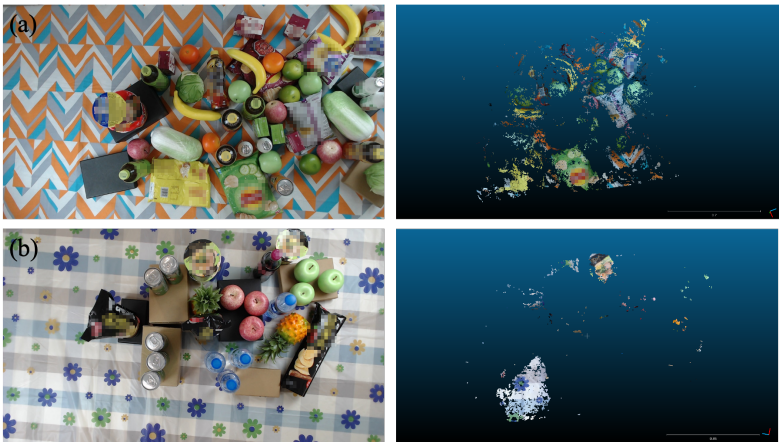

Fig. 8: (a) and (b) are two examples of 3D reconstruction results: view #1 of the scene is placed on the left and the construction result on the right. SfM is performed by Theia [10] and multi-view stereo is performed by OpenMVS [2]

Fig. 9: SIFT keypoints have an imbalanced distribution among instances. There are instances with few keypoints, e.g., the yellow cup in the image

9.5 Visualization of SIFT Keypoints

We visualize the keypoints detected by SIFT, as shown in Figure 9. It is clear that SIFT keypoints cluster at feature-rich regions such as edges and patterns. Texture-less instances, however, have very few keypoints. This imbalanced distribution of keypoints is likely the reason for the poor performance.

References

1. Bradski, G.: The OpenCV library. Dr. Dobb's Journal of Software Tools (2000)
2. Cernea, D.: OpenMVS: Open multiple view stereovision (2015)
3. Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F.J., Medina-Carnicer, R.: Generation of fiducial marker dictionaries using mixed integer linear programming. Pattern Recognition (2016)
4. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge University Press (2003)
5. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. PAMI (2007)
6. Moulon, P., Monasse, P., Perrot, R., Marlet, R.: OpenMVG: Open multiple view geometry. In: International Workshop on Reproducible Research in Pattern Recognition (2016)
7. Romero-Ramirez, F.J., Muñoz-Salinas, R., Medina-Carnicer, R.: Speeded up detection of squared fiducial markers. Image and Vision Computing (2018)
8. Schönberger, J.L., Frahm, J.M.: Structure-from-Motion Revisited. In: CVPR (2016)
9. Shorten, C., Khoshgoftaar, T.M.: A survey on image data augmentation for deep learning. Journal of Big Data (2019)
10. Sweeney, C.: Theia multiview geometry library: Tutorial & reference. http://theia-sfm.org
11. Wei, X.S., Cui, Q., Yang, L., Wang, P., Liu, L.: RPC: A large-scale retail product checkout dataset. CoRR **abs/1901.07249** (2019)
12. Winder, S., Hua, G., Brown, M.: Picking the best DAISY. In: CVPR (2009)